

For Tandy's 1000 and 600 Portables and 1000, 1200, 2000 and 3000 MS-DOS Computers

PCM

The Personal Computer Magazine
for Tandy® Computer Users

Vol. III No. 8
February 1986
U.S. \$3
Canada \$3.95

Towering Database Management From Subroutine City



**Upgrade Your
Tandy 1000**

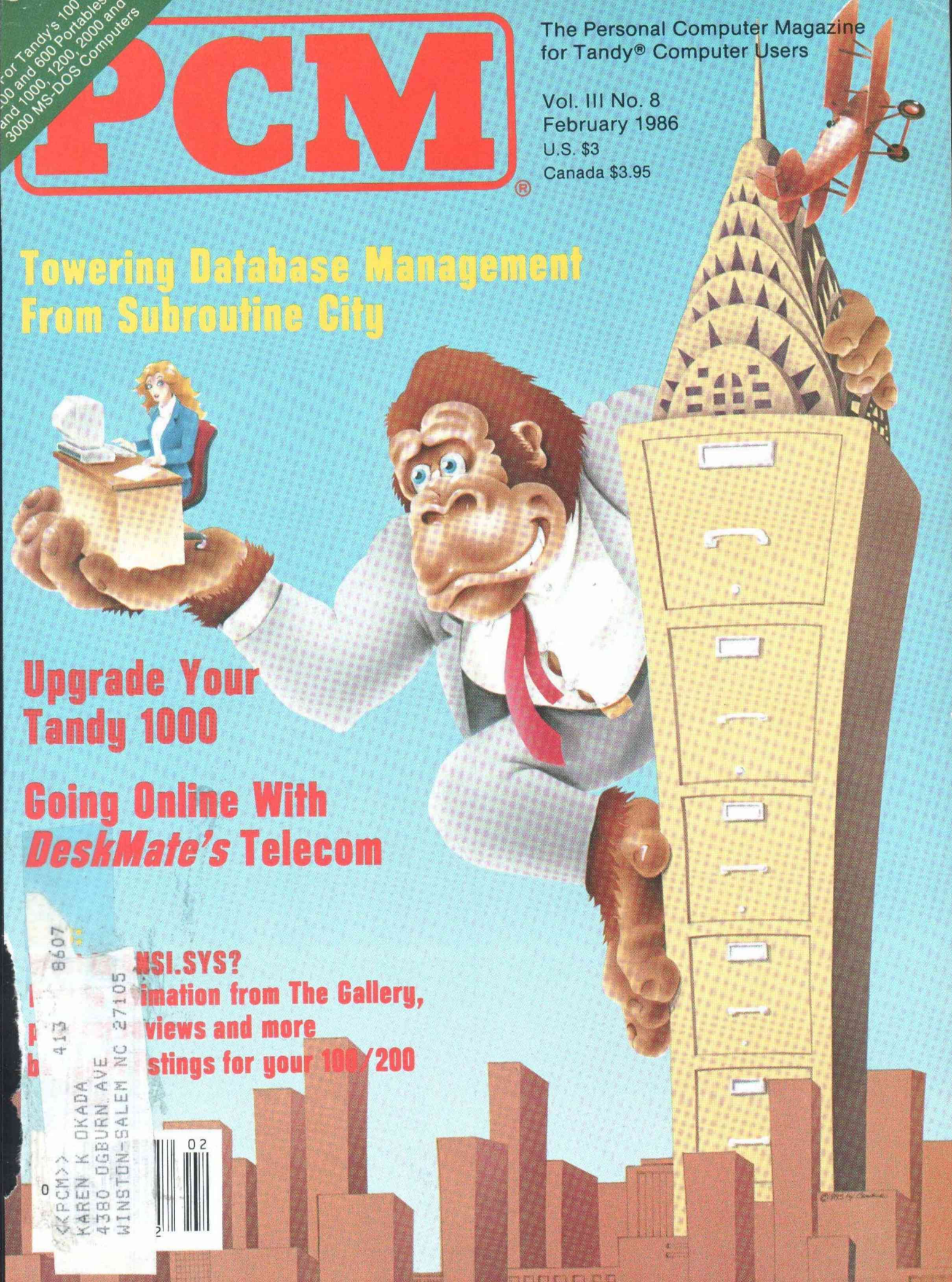
**Going Online With
DeskMate's Telecom**

NSI.SYS?
Information from The Gallery,
views and more
strings for your 100/200

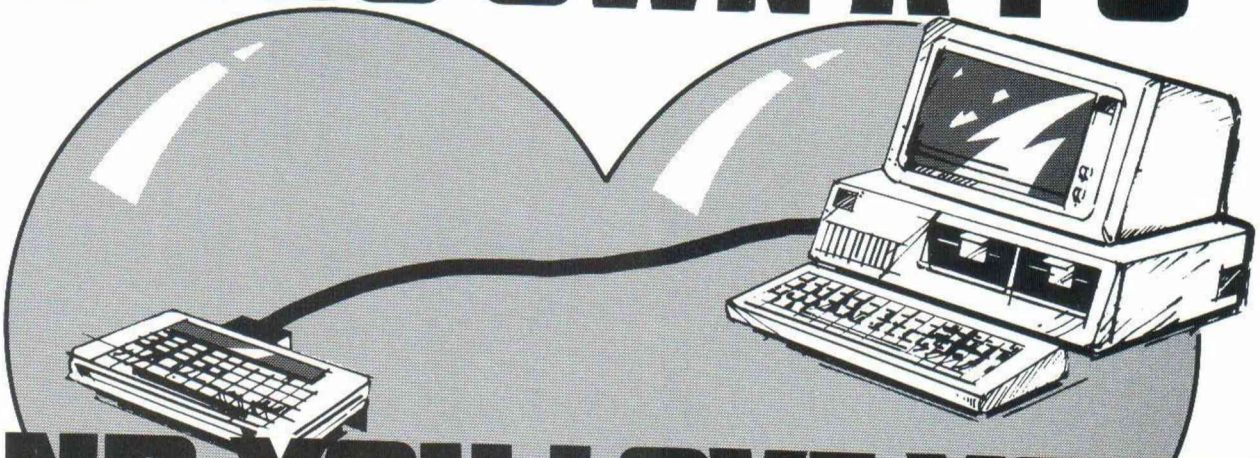
<<PCM>>
KAREN K OKADA
4380 OGBURN AVE
WINSTON-SALEM NC 27105



02



IF YOU OWN A PC



AND YOU LOVE YOUR MODEL 100 OR TANDY 200...

Call This Number Now 1-617-647-1099

Let Telecommuter software turn your IBM-PC into an integrated word processing and communications system as powerful as any combination of software products you can find.

Our software is as easy to learn and use as your portable computer. If you can use a Model 100 or Tandy 200, you already know how to use Telecommuter on your personal computer. It's that simple!

Telecommuter runs on the IBM-PC/XT/AT/jr and compatible MS-DOS computers, such as Compaq, AT&T 6300 and Tandy 1000/1200/2000/3000.

Start a job on your portable, then use Telecommuter to move the files to your desktop and finish the work with big computer word processing and communications power — FAST!

Call today to order your copy!

TELECOMMUTER™

*Software that always works
Support that's always there*

Only \$200. MasterCard and Visa accepted.

SO SIMPLE

SIGEA SYSTEMS, INC.

19 Pelham Road, Weston, MA 02193 (617) 647-1099

Basic.

Unfortunately, a lot of Word Processing Software is not what it seems. It is not easy. But it is expensive.

Leading Edge wants to change the way you think about Word Processing.

By making it easy *yet* inexpensive.

Quite simply, we've taken advantage of all the features of personal computers like the IBM[®], Tandy[®] and Leading Edge[®] PCs and made them appealingly simple to use.

Even if you're a two-finger typist. Even if you've never used word processing before in your life.

Easy. Simple. A snap. Duck Soup. There are no codes to memorize. No complex functions. And the system guides you all along the way, in plain and simple English.

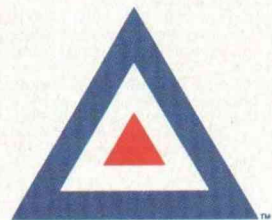
What's the catch?

There is none. Leading Edge Software Products go from Basic to Brilliant. And then, Beyond. With the kind of support that everybody else promises, yet rarely delivers.

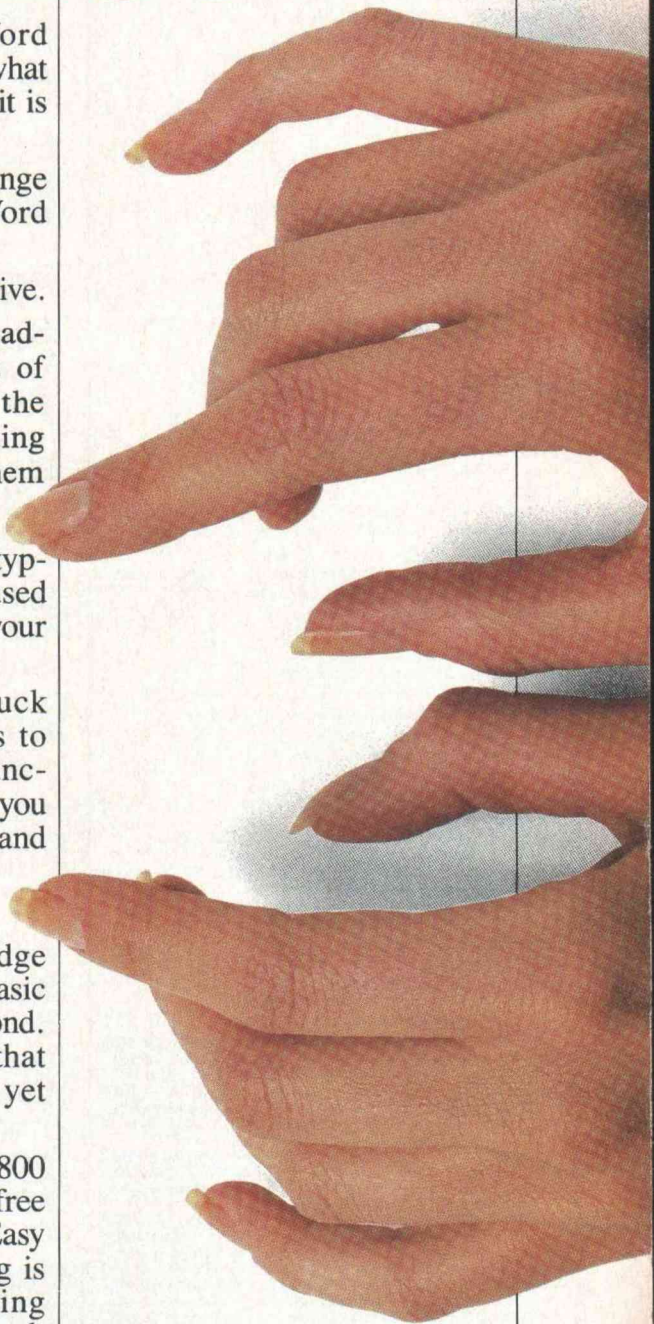
To find out more, just call 800 523-HELP and ask for our free brochure. It's easy to get. Easy to read. The only hard thing is understanding how Leading Edge Products can do so much for you, so fast, for so little.

LEADING EDGE[®]
Word Processing

IBM is a registered trademark of International Business Machines.
Leading Edge is a trademark of Leading Edge Products, Inc.



Leading Edge Software Products, Inc.
21 Highland Circle, Needham Heights, MA 02194
1-800-523-HELP, (617) 449-4655



Editor and Publisher
Lawrence C. Falk

Managing Editor Danny Humphress
Submissions Editor Jutta Kapfhammer
Copy Editor Tamara Renee Dunn
Reviews Editor E. Monica Dorth
Editorial Assistants Wendy Falk, Judi Hutchinson,
Shirley Morgan
Technical Assistants John R. Curl, Ed Ellers
Contributing Editors William Barden, Jr.,
Jim Hawk, Richard White
Consulting Editors Belinda C. Kirby, T. Kevin Nickols

Art Director Tracey Jones
Designers Heidi Maxedon, Kevin Quiggins
Sandra Underwood
Production Assistant Cindy Jett

Chief of Typography Debbie Hartley
Typography Services Jody Doyle, Angela Kapfhammer,
Suzanne Benish Kurowsky

Falsoft, Inc.

President Lawrence C. Falk

General Manager Patricia H. Hirschi

Editorial Director James E. Reed
Asst. Editorial Director Jutta Kapfhammer
Executive Editor Courtney Noe
Creative Director Jerry McKiernan
Manager of Public Relations Holly Weaver

Administrative Assistant to the Publisher Judy Brashear
Asst. General Manager for Finance Donna Shuck
Bookkeeper Diane Moore
Advertising Accounts Beverly Taylor
Dealer Accounts Judy Quashnock
Accounts Payable/ADP Lisa Ragan
Director of Fulfillment Services Bonnie Frowenfeld
Asst. Dir. of Fulfillment Services Sandy Apple
Asst. Customer Service Manager Beverly Bearden
Customer Service Representative Monica Wheat
Word Processor Manager Patricia Eaton
Chief of Production Services Melba Smith
Dispatch Janice Eastburn
Business Assistants Laurie Falk, Pam Workhoven,
Sharon Smith

Advertising Coordinator Doris Taylor
Advertising Assistant Debbie Baxter
(502) 228-4492

Advertising Sales Representatives
Kim Vincent, Cindy Shackelford

For PCM Advertising
and Marketing Office
Information, see Page 98

PCM — The Personal Computer Magazine for Tandy®
Computer Users is published every month of the year by
FALSOFT, INC., The Falsoft Building, P.O. Box 385,
Prospect, KY, 40059. Phone (502) 228-4492. PCM — The
Personal Computer Magazine for Tandy® Computer Users
and the PCM logotypes are registered ® trademarks of
FALSOFT, Inc.

Second class postage paid Prospect, KY, and additional
offices. USPS N 713-470 (ISSN 0747-0460). POSTMASTER:
Send address changes to PCM, P.O. Box 385, Prospect, KY
40059. Forwarding Postage Guaranteed.

Entire contents copyright © 1986 by FALSOFT, Inc. PCM
— The Personal Computer Magazine for Tandy® Computer
Users is intended for the private use and pleasure of its
subscribers and purchasers and reproduction by any means
is prohibited. Use of information herein is for the single end
use of purchasers and any other use is expressly prohibited.
All programs herein are distributed in an "as is" basis,
without warranty of any kind whatsoever.

TRS-80, Model 100, Tandy 200, 600, 1000, 1200 HD, 2000,
3000 and Portable Computer are registered ® trademarks of
the Tandy Corp. CompuServe is a registered ® trademark
of CompuServe Inc.

Subscriptions to PCM — The Personal Computer Maga-
zine for Tandy® Computer Users are \$28 per year in the
United States. Canadian rates are U.S. \$35. Surface mail to
other countries is U.S. \$64, air mail U.S. \$85. All subscrip-
tions begin with the next available issue.

Payment accepted by VISA, MasterCard, American
Express, Cash, Check or Money Order in United States
currency only.

Limited back issues are available. Please see notice for issues
that are in print and their costs. Payment accepted by VISA,
MasterCard, American Express, Cash, Check or Money Order in
U.S. currency only.

Features

Electric Blackboard/ <i>Leonard Hyre</i>	10
An entertaining math tutorial program	
The Integrated Desk/ <i>Bobby Ballard</i>	20
Explore DeskMate's Telecom autolog procedures	
Easy Errors/ <i>John & Aileen Cornman</i>	23
Adds easy-to-understand error messages to Model 100 BASIC	
Upgrade your Tandy 1000/ <i>Brian Alsop</i>	25
A do-it-yourself guide for adding memory and drives	
Windchill/ <i>Wayne Sanders</i>	36
Use this BASIC program to compute the windchill factor	
Delphi Bureau/ <i>Kevin Nickols</i>	38
What's happening this month in PCM's Delphi connection	
DOS Boot/ <i>John B. Harrell, III</i>	44
What is ANSI.SYS?	
The Gallery/ <i>Wayne Sanders</i>	49
3-D graphics and palette animation	
Emergency Power for your Tandy 600/ <i>Carl Oppedahl</i>	52
Build this simple and essential backup power supply	
PCMfest Reporter/ <i>Jeffrey S. Parker</i>	55
News from PCM's first computer show	
Subroutine City/ <i>Robert Covington</i>	57
A towering database system for your MS-DOS computer	
Barden's Buffer/ <i>William Barden, Jr.</i>	67
A fair comparison of BASIC, PASCAL and C	
Printer Setup Utility/ <i>Marshall K. DuBois</i>	78
Use this BASIC program to configure your printer	

Departments

Advertisers Index	130
Back Issue Information	121
Bar Coded Listings	126
Letters to PCM	8
Lprint/ <i>Lawrence C. Falk</i>	6
Editor's Notes	
New Products	125
The Rackseller	130

Reviews

Diagram Master/ <i>DecisionResources/1000, 1200, 2000</i>	120
Gramarcy/ <i>Cheapware/2000</i>	123
Model 100/200 Disk Drive/ <i>Tandy/100, 200</i>	124
Offix/ <i>Tandy/1000, 1200, 2000, 3000</i>	115
Pop-Up DeskSet/ <i>Bellsoft, Inc./1000, 1200, 2000</i>	109
PowerText Formatter/ <i>Beaman Porter, Inc./1000</i>	117

Tandy computers offer everything ...almost

Check out the software that completes their systems
—**filePro®**

Tandy MS-DOS Software Comparison Chart

	pfs® File/Report	dBase™ II	filePro 16 Smallware
GENERAL CHARACTERISTICS:			
Menu driven	yes	no	YES ✓
Allows user to create integrated business systems	no	programmer required	YES ✓
Developed systems and data can be moved to multi-user environments	no	no	YES ✓
Professional support available from the software's authors	no	no	YES ✓
PRICE	\$265	\$595	\$495
CAPACITIES:			
Fields per record	100	32	999 ✓
Characters per record	1679	1000	4608 ✓
Records per file	1300	65535	16,000,000 ✓
Indexes per file	1	7	12 ✓
Number of digits per numeric field	20	10	24 ✓
Number of files usable concurrently	1	2	10 ✓
Files span multiple drives	no	no	up to 8 ✓
FEATURES:			
Full-screen facility for creating custom screen layouts	yes	no	YES ✓
Full-screen facility for creating custom report layouts	no	no	YES ✓
Built-in field types (error checking)	no	3	12 ✓
User-defined field types	no	programmer required	200 ✓
Conditional math	no	programmer required	YES ✓
User-defined menus	no	programmer required	YES ✓
Change file layout without losing existing data	possible	possible	automatic ✓
Data protection	no	programmer required	YES ✓
Password security	no	programmer required	YES ✓

The new Tandy computers can really speed things up. If you don't let obsolete software bog you down. Try *filePro 16*, the latest generation of data base software from the authors of *Profile®*.

Choose it for its power (check our capacities); its ease-of-use (check our features); and its unique ability to grow as your needs grow, and even move the systems you create to multi-user computers.

filePro is available for the Tandy 1000, 1200 and 2000 computers (hard disk required). Available directly from us or through your Radio Shack store's Express Order Software service.

**To order, call
(800) 847-4740.**

For customer service or when dialing in New York State, call (212) 398-9290. Ask for Smallware sales. Visa and MasterCard accepted.

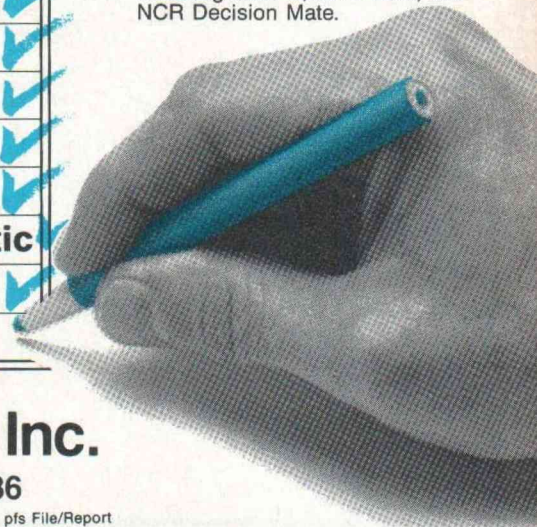
Also available for other computers, including IBM PC, IBM PC-XT, and NCR Decision Mate.



The Small Computer Company, Inc.

230 West 41st Street, Suite 1200, New York, New York 10036

Smallware and *filePro* are registered trademarks of The Small Computer Company, Inc. dBase II is a trademark of Ashton-Tate. pfs File/Report is a registered trademark of Software Publishing. Profile is a registered trademark of Tandy Corporation.





First Weeks, Final Touches

We have developed a number of new friends here at PCM this month. They are Jeff and Rodney, Rick and Sam. And, of course, Dana.

For the past several weeks, these people — and many more — have been living with us in the new Falsoft Building. They are the people who are doing the final touches on our building and getting things pretty much in shape.

Jeff, who prefers the appellation “Sparky,” is our “on site” electrician. Dana is the contractor who supervises this band of merry men.

Moving from 4,800 square feet to 18,000 on three floors has required some adjustment. Thanks to Jim, things have been easier. He installed our new telephone system — so when we need Rodney, we just use the phones to page a message like “desperately seeking Rodney” rather than go hunting for him.

I guess we’ve been a good influence on everyone, too. Several of our new friends have bought Tandy computers since the job started!

All of this is to say that we’ve *finally* moved into our new building. The kinks are certainly not out yet, by any means, but we are here, we’re warm (once they fixed the heat), and everyone is settling in.

Considering that, before the move, each staff member had an average of 96 square feet in which to work and now each has 360, you can imagine there is a world of difference in efficiency alone. Each of our editors, for instance, has a countertop for his or her computer(s), printers and whatever; shelves above the counters for software, manuals and the like; and, (hurrah!) electric outlets and power strips galore.

We even have enough telephone lines that we are able to have several people on modems at the same time (a big boost for our Delphi activities), room to have board sessions in our own conference room (away from the hubbub) and so on.

We’ll have an “open house” after the first of the year and we want you to come! If you can’t make it, try to make Prospect (Louisville) a stop on your next trip. We’d love to show you around.

COMDEX was an exciting show for us. One of the stars of the show was the new Tandy 3000, and the 600, too, though a little less so. Both, of course, were featured on December’s cover. And there was a lot of talk at the show about Tandy, which boasted a sensational new booth as well as super new machines.

Some of the “reviews” of COMDEX said that it was “dull,” but I think that those who said that were, generally, misinformed. The computer industry has matured to the point where you don’t need dancing elephants for excitement. Rather, simply by working the show, you’ll find (I know I did) a lot of exciting things happening. Things you can use on and with your Tandy computer.

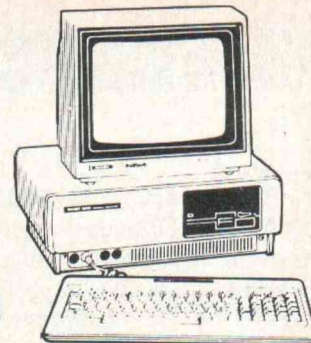
Let’s not mistake hype and hoopla for progress and growth.

Finally, Eric Maloney of *80 Micro* informs me I was wrong last month about John Roach’s picture appearing on PCM’s cover as being a “computer magazine first.” Eric’s own magazine beat us to that distinction, as did *InfoWorld*.

Oh, well.

— Lonnie Falk

PERRY COMPUTERS



CALL

1-800-248-3823

COLOR COMPUTERS

	LIST PRICE	OUR PRICE
26-3127 64K Extended Color Computer 2	\$199.95	\$175.00
26-3134 16K Standard Color Computer 2	119.95	100.00
26-3129 Thinline DD O for Color Comp	349.95	290.00
26-3018 Extended Basic Kit	39.95	36.00
26-3030 OS-9 with Editor Assembler	69.95	59.50
26-3012 Deluxe Joystick	29.95	25.00
26-1208 CCR-81 Recorder	59.95	50.00
VIP Integrated Library	149.95	139.00
VIP Writer	69.95	59.00
VIP Calc	69.95	59.00
Teletwriter Disk		49.00
Botek Interface		59.00

TANDY 3000

25-4001 Tandy 3000 1 Disk 512K	\$2599.00	\$1900.00
25-4010 Tandy 3000 1 FD 20 Meg HD 512K	3599.00	2600.00
25-3046 Deluxe Text Display Adapter	249.95	205.00
26-3047 Deluxe Graphics Display Adapter	499.95	395.00
25-4030 512K to 2Meg Expansion Board	499.00	395.00
25-4033 Math Co-Processor for 3000	399.95	339.00
25-4050 1.2 Meg Disk Drive Kit for 3000	299.95	255.00
25-4060 HD Controller Board for 3000	499.00	425.00
25-4062 20 Meg Hard Disk Kit	799.00	679.00
25-4101 MS-DOS 3.1/Basic/Deskmate	99.95	85.00
26-5111 Monochrome Monitor	199.95	165.00
26-5112 Color Monitor	599.95	510.00

TANDY 100, 200, 600

26-3901 Tandy 600 Port Comp 32K 3 1/2 DD	\$1599.00	\$1125.00
26-3904 Basic Rom for Tandy 600	129.95	110.00
26-3910 92K RAM Upgrade for Tandy 600	399.95	340.00
26-3860 Tandy 200 Port Comp 24K	999.00	725.00
26-3866 24K RAM Mem. Exp. Chip for 200	249.95	210.00
26-3802 Tandy 100 Portable Computer 24K	499.00	425.00
26-3808 Tandy 100/200 3 1/2" Drive	199.00	169.00
26-3816 8K RAM Upgrade for Tandy 100	119.95	95.00
26-3909 Portable 3 1/2 DD for T100 T200	199.95	169.00
26-3805 Acoustic Coupler	39.95	34.00
26-1409 Printer Cable	14.95	12.70
26-1410 Modem Cable	19.95	17.00

MODEL 4D

26-1070 Model 4D 64K 2 FD w Deskmate	\$1299.00	\$ 895.00
26-1122 64K Memory Expansion	69.95	65.00
26-1123 Model 3 to Model 4 Upgrade Kit	399.00	385.00
26-1134 Model 4 Hard Disk Kit for 15meg	79.95	67.95
26-1530 Multiplan Model 4	199.00	169.00
26-1595 Super Scripts Model 4	199.95	169.00
26-1608 Deskmate for Model 4	195.00	175.00
26-1635 Profile 4 Model 4	249.95	200.00
26-2216 CP/M Plus Model 4	149.00	127.00
26-2231 Double Duty Utility	69.95	58.00

MODEL 6000 COMPUTERS

26-6021 Model 6000 512K 2 FD Computer	4499.00	3125.00
26-6022 Model 6000 512K 1 FD 15Meg HD	5499.00	3795.00
26-6019 Model 6000 256K Memory Kit	199.95	185.00
26-6052 DT-100 Data Terminal	795.00	600.00
26-1245 10 Meg Disk Cartridge System	2195.00	1750.00
26-6018 6000 Interface Kit for 26-1245	119.95	102.00
25-3022 Interface Kit for 26-1245	199.95	160.00
26-4155 15 Meg Hard Drive Primary	1995.00	1355.00
26-4171 35 Meg Hard Drive Primary	2995.00	2545.00
26-4173 Meg Hard Drive Primary	4295.00	3350.00
26-4157 Installation Kit for Primary HD	349.00	295.00

All prices and offers may be changed or withdrawn without notice. Advertised prices are cash prices. C.O.D. accepted (\$10.00 charge per carton on C.O.D. Call for further C.O.D. information.) M.C., Visa, add 2%. AX, add 3%. All non-defective items returned will be subject to 10% restocking fee. Defective items require return merchandise authorization. Call for R.M.A. Number before returning. Delivery is subject to product availability.

RADIO SHACK PRINTERS

	LIST PRICE	OUR PRICE
26-1276 DMP-105 Dot Matrix Printer	\$ 199.00	\$ 169.00
26-1275 TRP-100 Portable Thermal Printer	299.00	230.00
26-1280 DMP-130 100 cps Tri. Mode Printer	349.00	285.00
26-1268 CGP-220 Color Ink-Jet Printer	699.00	595.00
26-1257 DWP-220 Daisy Wheel	599.00	460.00
26-1277 DMP-430 24 Wire Matrix Printer	899.00	635.00
26-1270 DWP-510 43 cps Daisy Wheel	1495.00	1225.00
26-1274 DMP-2100P 24 Dot Wire Matrix	1995.00	1650.00
26-1279 DMP-2200 HiSpeed Matrix Printer	1695.00	1440.00
26-1269 PT-64 Printer Controller	249.95	210.00
26-1498 SW-302 Printer Switch	119.95	100.00
26-1477 Auto Sheet Feeder for DWP 510	499.95	420.00

TANDY 1200, 2000

25-3000 Tandy 1200 1 FD & 10 Meg HD	\$1999.95	\$1525.00
25-3001 Tandy 1200 2 FD 256K	1499.00	1200.00
25-3010 VM-3 Green Monitor	219.95	185.00
26-3212 CM-2 Color Monitor	459.95	390.00
25-3043 Graphics Display Adapter	299.95	255.00
25-3044 Graphics Master	695.95	540.00
25-3061 Captain Multifunction Board	599.95	475.00
25-3020 TCS-100 Tape Cartridge System	1999.00	1555.00
25-3021 Tandy 1000/1200 Interface Kit	149.95	120.00
25-3130 MSDOS/BASIC	89.95	76.50
26-5103 Tandy 2000 Two Disk	1599.00	1225.00
26-5104 Tandy 2000 1 FD and 10 Meg HD	2499.00	1885.00

TANDY 1000

25-1000 Mod 1000 128K 1 FD & Deskmate	\$ 999.95	\$ 705.00
25-1001 Model 1000 256K 1 FD 10Meg HD	1999.00	1475.00
25-1003 300-Baud Modem Board	149.95	125.00
25-1005 Disk Drive Expansion	199.95	170.00
25-1006 RS-232C Interface 1000/1200	99.95	85.00
25-1007 Hard Disk Control Board	299.95	255.00
25-1013 1200-Baud Modem Board	299.95	230.00
25-1025 10 Meg Hard Disk Drive System	699.95	559.00
25-1501 MS-DOS Reference Manual	34.95	29.00
25-1502 BASIC Reference	34.95	29.00
26-1145 Lotus 123 1000	495.95	420.00
30051030 PJB Multifunction Board 512K		310.00

EPSON PRINTERS

20001015 Homewriter 10	\$ 288.00	\$ 215.00
20001025 LX-80 Dot Matrix Printer	299.00	225.00
20001035 FX-85 Dot Matrix Printer	499.00	385.00
20001040 JX-80 Color Dot Matrix Printer	699.00	485.00
20001050 HI-80 4 Pin Plotter	599.00	390.00
20002010 RX-100 Dot Matrix Printer	499.00	400.00
20002030 FX-185 Dot Matrix Printer	699.00	525.00
20003010 LQ-1500 18 Pin Head with Intf	1495.00	1100.00
20001515 LX-80 Tractor Feed	49.00	25.50
10081010 DX-10 Daisy Wheel Printer	299.00	255.00
10081020 DX-20 Daisy Wheel Printer	459.00	390.00
10081030 DX-35 Daisy Wheel Printer	899.00	765.00

For Technical Questions and Information on our complete line of computer accessories and current prices.

CALL 1-517-625-4161

FOR ORDERS ONLY CALL 1-800-248-3823

Mon., Wed. & Fri. 9-9, Tues. & Thurs. 9-6, Sat. 9-3

124 S. MAIN ST, PERRY, MICH. 48872

Editor:

I have a Tandy 1000 and a CGP-220 printer. The December 1985 issue has a program in it which I entered into my computer (The Gallery, Page 52). It worked on the computer screen but would not print on the printer.

Debbie Kimler

Editor's Note: This program was not written to automatically print on your printer. You must have the DUMPCGP .SYS driver installed in your CON FIG.SYS file. Once the image is displayed on the screen, press SHIFT-PRINT and it will be dumped to your printer.

To install the screen print driver in your configuration, type the following at the MS-DOS prompt, pressing ENTER after each line:

```
COPY CONFIG.SYS+CON CONFIG.SYS
DEVICE=DUMPCGP.SYS
CTRL-Z
```

You will have to reset the system with this disk for the driver to be installed.

Editor:

I just discovered that *Handi 1000* (January 1986) will not run on a 128K machine as published. The 128K machine will not support SCREEN 6 and the large program. By way of amends, I am including the needed changes to make a SCREEN 2 functioning version for 128K users. It probably will run on Tandy 1200, 2000 and 3000 also!

Type in the modified lines, save them as an ASCII file and merge them with the original to get a working copy.

Leonard Hyre
Cambridge, MD

*Editor's Note: To modify Handi 1000 for the 128K Tandy 1000 do the following:
This version should run on 1200 and 3000 also.*

Delete lines 110, 120 and 830.

Change the following lines as shown or type them in separately and type SAVE "MOD.BAS", A to save the new lines as an ASCII file. Next, load HAND1000 and type MERGE "MOD.BAS". Then

save the whole thing as HAND 128.BAS.

```
110 KEY OFF:SCREEN 2,1:CLS
160 CLS:GOSUB 1720:PRINT STRING$(80,"*"):LOCATE 3,29:PRINT
  "The. . . .H A N D I - 1000":
  LOCATE 5,1:PRINT STRING$(80,"*")
410 CLS:LINE(380,80)-(600,
  130),1,B:LINE(440,100)-(470,
  128),1,B:LINE(525,100)-(57,
  118),1,B
940 CLS:PRINT STRING$(80,
  202):LOCATE 2,1:PRINT"
  WALLPAPER CALCULATION":
  LOCATE 3,1:PRINT STRING$(80,202)
1240 CLS:PRINT STRING$(80,
  "4"):IF AN$="4"THEN SY$
  ="CARPETING A ROOM" ELSE IF
  AN$="6"THEN SY$="* TILING
  A ROOM *"
1270 LINE(360,140)-(620,
  190),1,B:LINE(450,140)-(500,
  156),1,BF:LINE(620,160)-
  (636,184),1,BF
1770 CLS
```

Editor:

I sold my Tandy 2000 and purchased a Tandy 1000 in order to be able to afford all "the goodies." I'm happy with the move, although I miss the very fast, high density disks, the large keyboard buffer and most of all, a readable screen display!

I noticed that the Tandy 1200 also has excellent text quality, probably matching that of the 2000's. How do I get it on my Tandy 1000? Why are the characters so fuzzy on the VM-2? Is it the character generator chip? Please help, I'm going blind.

W. Aird Flavelle
Calgary, Alberta

Editor's Note: The Tandy 1000 provides two monitor outputs. One for an RGBI (Red-Green-Blue-Intensity) monitor and another for a "composite video" monitor. The VM-2 connects via the composite jack, which produces a signal similar to what you would get on the video output jack of your home VCR. The Tandy 1200 and 2000 use a different technique for their monochrome display, TTL signals. This type of

interface, designed for this computer applications — much higher-resolution displays.

In short, the problem is with the computer — not the monitor. The only solution is to switch to a high-resolution RGBI monitor.

Editor:

I have been operating a Tandy 2000 at our business for about a year-and-a-half now. I subscribe to your excellent magazine, *80 Micro* and *Business Software*. It seems there is very little information regarding homebrew interfaces and improvements for users.

For example, Tandy lists the 10 Mb hard disk drive assembly for \$1,300+. If we Tandy lovers had IBM PCs we could mail order a similar hard disk drive for about \$700! I wonder if there's any feasible way to use an independent supplier's hard disk package on the 2000? Undoubtedly, the controller card wouldn't fit precisely but, would the electronics be right? Or, can a person buy the controller card from Tandy by itself and supply another drive?

Another titilating item is the abundance of cheap RAM chips available by mail order. What kind of chips does the Tandy 2000 use in the memory expansion boards; 64K or 256K? What happens if you buy an expansion board and decide to populate it with 256K RAM chips? Please explain the differences between these chips and what can be done with them.

I am curious about the 640K RAM board advertised in PCM. Does it perform as advertised? I made an inquiry to the manufacturer but have not received a reply.

I believe that other readers have similar questions and would appreciate seeing your comments in print.

Lew A. Maneval
Jasper, MO

Editor's Note: Tandy does not sell their hard disk controller board without the disk drive, so you pretty much have to stick with their package.

You can, however, install your own 64K RAM chips in the Tandy expansion boards. 256K chips would not work in these boards, though, because the boards were not designed to address the higher-density chips.

TANDY

COMMODORE

IBM PC/XT/AT SYSTEMS

AND TANDY COMMODORE

POWER

PCM**IS NOW
ON DISK**

**\$99 for a year's
subscription to
PCM ON DISK!**

Each and every month, PCM offers an assortment of programs for the home and office — utilities, business applications, games and more — all for the price of typing in the listings.

If you're a person on the move, and too busy to spend time typing lengthy listings, then PCM has just what you need —

PCM ON DISK.

PCM ON DISK provides each issue's MS-DOS programs ready to load and run with only a few keystrokes!

AS AN ADDED BONUS, **PCM ON DISK** will include PCM's portable computer programs, which may be downloaded to portables for those readers who have both an MS-DOS machine and one of Tandy's portable computers.

That means you get a disk each month for a year, containing all programs listed in that month's issue, for an average cost of only \$8.25!* We think it's worth your time to subscribe.

*Disks for individual issues (beginning this month) may be ordered for only \$12 each.

PCM**PCM ON DISK**

The Falsoft Building
P.O. Box 385
Prospect, KY 40059
(502) 228-4492

I would like a one-year (12 months) subscription to PCM ON DISK.

Name _____

Address _____

City _____ State _____ ZIP _____

Payment Enclosed ☐

Charge Visa ☐ MasterCard ☐ American Express ☐

My Account# _____ Card Exp. Date _____

Signature _____

*Subscriptions to PCM ON DISK are \$99 a year in the United States. Canadian rate U.S. \$115. All other countries U.S. \$115; PCM ON DISK single copy rate is: \$12 within the U.S.; U.S. \$14, Canadian rate; U.S. \$16, all other countries. Kentucky residents add 5% sales tax. All subscriptions begin with the current issue unless otherwise specified. Please allow 5 to 6 weeks for first copy. U.S. currency only, please. In order to hold down non-editorial costs, we do not bill.

This computerized
"blackboard" will lead
the student through the
logical steps of problem
solving

The Electric Blackboard

By Leonard Hyte

here are many reasons for praising the virtues of owning a computer. Among these, one reason that seems to have the greatest possibility for home use is that of home education. The Tandy 1000 possesses many of the attributes needed to excel in this field. Excellent graphics capability, large amounts of available memory and a flexible language base are all there.

Math drill programs abound for the student, many of them excellent programs. But it seems that programmers tend to miss one important area of the advancing math student. When a student is first faced with the need to work multiplication and division problems with larger numbers, the concepts of carrying remainders and the need to perform multiple functions on one problem are difficult to grasp. This program is an attempt to aid these students, leading them through each logical step required to solve a given problem.

In the *Electric Blackboard*, the student is shown the problem as it would appear on a blackboard (would you

believe a white board). He or she is then led through the step-by-step solution of the entire problem. I believe this graphics demonstration of the logic will help the student grasp the concepts involved more quickly.

The *Electric Blackboard* can also serve as a programming exercise for the novice BASIC programmer, as it incorporates many of the excellent graphics and text capabilities of Tandy 1000 BASIC. There is nothing, however, that would go beyond the grasp of a novice BASIC programmer. Also, I must point out for those of you with "Big Blue" PCs, I'm afraid your machines are just not *Tandy* compatible enough for the BASIC graphics capability of the 1000.

Electric Blackboard can be thought of as consisting of four separate main routines plus a couple of smaller side-tracks for housekeeping and such. The four main sections are Title Screen, Classroom Scene, Multiplication and Division. Additionally, the smaller "Turn out the lights" routine is separate.

I would like to discuss each section separately, but would first like to give you some of the generalities. The program uses SCREEN 6, allowing us to use Hi-Res graphics and text on the same screen. If the 1000 is your first machine, you may not realize what a unique capability this is. To set aside the memory required for SCREEN 6, we must first CLEAR , , , (32768). In other words, tell BASIC that this graphics screen is going to need that much memory saved for our program's graphics. Note that when we type in the line

(Leonard Hyre works as a claims representative for the Social Security Administration. He has written several articles for THE RAINBOW, PCM's sister publication for the Color Computer, and is the author of a number of commercial programs. He may be contacted at P.O. Box 403, Cambridge, MD 21613; 301-228-0064.)

(Line 120), we type CLEAR ..., (32768). BASIC changes it to CLEAR ..., (32768!). You do *not* type in the Exclamation Mark!

The title screen is strictly window dressing; that is, the program would function fine without it. However, I have a weakness for such things and the vision of a blackboard discharging bolts of electricity was irresistible. Line 140 sends the computer all the way to 1940 to put the title on the screen and give the author his ego strokes. Nothing fancy here; the bolts of electricity are simply drawn in yellow (Color 1... since we used Palette 1,14 to make it so) and then overdrawn in black to make them disappear. After a few seconds of this we return to the GOSUB at 140 and continue with the next line.

The next step in program operation is the creation of the classroom scene with a brick wall, teachers desk, a "white" blackboard and a bit of decor. When typing in the STRINGS from which the wall is created, you simply press the ALT key and, while holding it down, use the numeric keys to type in the ASCII number of the character needed. Please refer to lines 1290-1310 of the program for more explanation of this method. We do not need to create great detail in our classroom. We need only to give an illusion of a classroom to set a proper mood. Perhaps the trickiest graphics command is PAINT, because the painting begins at the specified point of origin and flows in every direction until the stated border color is reached. A bit of a leak, and you end up with a solid color screen. The screen finished, the program jumps back to the GOSUB from which it came.

The RETURN takes us back to 1220, where the program picks up another GOSUB to Line 1730. You may wonder

why we needed to return in the first place. If you examine the two subroutines, you will find that the first, at lines 1220-1700 is the classroom, which is only needed one time for each time the program is run. The other is the choice menu, which is repeatedly called during program execution. We would not want to redraw the entire screen just to get another problem. Thus it is necessary to have separate routines.

Finally, the meat of the subject — mathematics. The multiplication is handled in the routine from 300-580, and the division is done from 630-1100.

From a programmer's point of view, the only tricky part here is manipulating numerics and strings to obtain the proper display and at the same time, keep track of the correct answer. Use of the powerful string-handling commands, RIGHT\$, MID\$ and STR\$, are used often. If you examine the routine line by line, the seemingly confusing combinations may soon make sense to you — a sure sign that "Computerese" is becoming your second language.

By the way, if you wish to change the difficulty level of the problems, you can do so by changing the values of A and

"We must determine the problem to be solved by selecting two random numbers, in this case a multiplicand in the range of 11 to 99 and a multiplier of less than 10"

Since these two routines share much the same type of programming problems, I will only detail one of them, the multiplication.

We must determine the problem to be solved by selecting two random numbers, in this case a multiplicand in the range of 11 to 99 and a multiplier of less than 10 (lines 320-340). The problem is then written on the blackboard in standard math notation; that is, the same way a student would write it out on paper. The student is then led through the problem. Wrong answers are not accepted, the student being forced to try again. This assures that the ultimate answer will be correct and the steps needed to arrive at that answer will have been performed by the student.

B, in lines 330 and 340 for multiplication and lines 650 and 660 for division.

If you have any problems using the *Electric Blackboard*, please feel free to write or call me. I am always glad to help with this type of problem. If the typing presents too much of a challenge to you, I will be happy to send you a copy of the program on disk. Just send me \$7 to cover costs and shipping. Of course, the program is also available from the MS-DOS SIG on DELPHI, at a nominal charge, and on PCM ON DISK.

My address is:

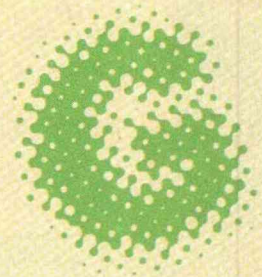
Leonard Hyre
P.O. Box 403
Cambridge, MD 21613
Phone (301) 228-0064 after 5 p.m.
EST. ☐

The listing:

```

20 *****
30 '*
40 '*          THE ELECTRIC BLACKBOARD
50 '*          by LEONARD HYRE
60 '*          (C) NOV. 1985
70 '*          For TANDY 1000 Users!
80 '*
90 *****
100 '
110 'NOTE: Do NOT Type in the '!' in the next line...BASIC WILL ADD IT

```



HARD DRIVE PRICE BREAKTHROUGH

STARTING AT \$299.95!

5 to 60 megabytes complete easy to install internal and external systems
— will boot just like an XT — fully warranted —
same day shipping from stock — fast service — free technical assistance

\$499.95 to \$999.95

STREAMING OR FLOPPY TAPE BACK UP SYSTEMS

complete internal and external systems

\$1,995.00 — IBM XT WORKALIKE SPECIAL

20 megabytes — one floppy — 640K — clock calendar — serial port — parallel port
monochrome card — monochrome monitor with tilt and swivel base
Keytronics 5151 style keyboard — one year warranty — shipped from stock

MEGADISK™ HARD DISK DRIVE SYSTEMS

TOLL FREE ORDERING 1-800-343-8841

For the { IBM/PC, Tandy 1000, TRS/80 Models I/III/IV/4P, Compaq, PC Workalikes, Color Computers, Heath/Zenith, Max/80

{ Complete with Hardware, Cables, Software and Quikfit Installation

5 Megabytes Internal Mount	starting at \$ 279.95
10 Megabytes Internal Mount	starting at 419.95
20 Megabytes Internal Mount	starting at 519.95
5 Megabytes External System	starting at 499.95
10 Megabytes External System	starting at 639.95
20 Megabytes External System	starting at 799.95
Tape Backup System — Internal; Or External	starting at 449.95

DOS Systems Available: { IBM/Heath — DOS, 1.0, 2.0, 2.1, 3.0, or later
{ TRS/80-LDOS, TRSDOS 6.x, Newdos/80, Dosplus, CP/M, COCO DOS, Max/80 LDOS, OS9

FULLY WARRANTIED — PARTS AND LABOR — 24 HOUR SERVICE — CALL TOLL FREE — 1-800-343-8841

IBM ENHANCEMENT CARDS

Multifunction with 384K	\$199.95	Color graphics	\$139.95
Serial port, parallel port		Monochrome/RGB Color	229.95
Clock calendar, game port		Disk controller with cable — up to 4 drives	79.95
Serial port	69.95	384K memory	129.95
Monochrome only	119.95	Hard drive controller	159.95
Monochrome graphics w/parallel port	139.95	Keytronics 5151 style keyboard	129.95
Network card			249.95

DEALER INQUIRIES INVITED

CALL 1-617-872-9090

PRICES CHANGE EVERY DAY

CALL 1-800-343-8841

PC

©
1985

SOFTWARE SUPPORT, INC.

1 Edgell Road, Framingham, MA 01701 (617) 872-9090 Telex-383425

Hours: Mon. thru Fri. 9:30 am to 5:30 pm (E.S.T.) Sat. 10 am to 3:30 pm

SERVICE POLICY — Our Professional Technical Staff Is Available To Assist You Monday Through Saturday.

WARRANTIES — Up To One Full Year Parts And Labor. Floppy Disk Drive Power Supplies — Five (5) Years.

SERVICE — 24 Hour Turn-A-Round On All In-Stock Parts. Dealer Inquiries Invited. Call 617-872-9090

Please Call For Shipping,
Handling And Insurance.
Cash Discounted Prices

Toll Free 1-800-343-8841

Please Call For Our Latest Price Saving Specials.

Not Responsible for Typographical Errors.
Prices and Specifications May Change
Without Notice.


```

120 CLEAR ,,,(32768!)
130 '
140 GOSUB 1940
150 '
160 '**** GO SET UP SCREEN ****
170 '
180 GOSUB 1220
190 '
200 '**** GO SET UP MENU ****
210 '
220 GOSUB 1730
230 '
240 '**** AWAIT MENU CHOICE ****
250 '
260 ON AN GOTO 300,630,1140
270 '
280 '**** MULTIPLICATION MAIN BODY ****
290 '
300 GOSUB 1810
310 LOCATE 4,6:PRINT"MULTIPLICATION Problems:"
320 RANDOMIZE
330 A=INT(RND(9)*99):A$=STR$(A):IF A<11 THEN 330
340 B=INT(RND(9)*7)+2:B$=STR$(B)
350 LOCATE 4,42:PRINT"THE PROBLEM IS: ";B" X "A
360 LOCATE 15,39:PRINT A:LOCATE 16,37:PRINT "X ";B:LOCATE 17,38:PRINT "áááá"
370 LOCATE 14,45:PRINT"Multiply ";RIGHT$(B$,1)" x ";RIGHT$(A$,1);" =";
380 C1=VAL(RIGHT$(A$,1))*VAL(RIGHT$(B$,1))
390 INPUT AN$
400 AN=VAL(AN$):IF AN<>C1 THEN LOCATE 14,61:BEEP:PRINT"      ":LOCATE 14,61:GOTO 3
90
410 GOSUB 1880:LOCATE 15,45:PRINT"WRITE IN ";RIGHT$(AN$,1);" and CARRY ";:IF C1<
10 THEN PRINT"0"; ELSE PRINT MID$(AN$,1,1);
420 LOCATE 18,41:PRINT RIGHT$(AN$,1):LOCATE 14,40:IF C1<10 THEN PRINT "0"; ELSE
PRINT MID$(AN$,1,1);
430 LOCATE 16,45:PRINT"Multiply ";RIGHT$(B$,1)" x ";MID$(A$,2,1);" =";
440 C2=VAL(RIGHT$(B$,1))* VAL(MID$(A$,2,1))
450 INPUT AX$
460 AX=VAL(AX$):IF AX<>C2 THEN LOCATE 16,61:BEEP:PRINT"      ":LOCATE 16,61:GOTO 4
50
470 GOSUB 1880:LOCATE 17,45:PRINT"ADD ";AX;" + ";:IF C1<10 THEN PRINT"0";ELSE PR
INT MID$(AN$,1,1);
480 IF C1<10 THEN AD=AX ELSE AD=AX+VAL(MID$(AN$,1,1))
490 INPUT AZ$
500 GOSUB 1880
510 IF VAL(AZ$) <> AD THEN LOCATE 17,45:BEEP:PRINT STRING$(19," "):GOTO 470
520 LOCATE 18,45:PRINT"WRITE IN ";AD:GOSUB 1880
530 IF A*B<100 THEN LOCATE 18,39:PRINT A*B;ELSE LOCATE 18,38:PRINT A*B
540 LOCATE 7,10:PRINT"CORRECT! HOW ABOUT ANOTHER (Y/N)?"
550 AK$=INKEY$:IF AK$=""THEN 550
560 IF AK$="Y"OR AK$="y" THEN 580
570 FOR WIPE=14 TO 19:LOCATE WIPE,37:PRINT STRING$(31," "):NEXT WIPE:GOSUB 1810
:RUN 220
580 FOR WIPE=14 TO 19:LOCATE WIPE,37:PRINT STRING$(31," "):NEXT WIPE:GOSUB 1880
:RUN 300
590 '
600 '

```



```

610 ***** DIVISION MAIN BODY *****
620 '
630 LOCATE 4,6:PRINT"DIVISION Problems:"
640 RANDOMIZE
650 A=INT(RND(9)*888)+100:A$=STR$(A)
660 B=INT(RND(9)*7)+2:IF B=< VAL(MID$(A$,2,1)) THEN 650
670 B$=STR$(B):LOCATE 4,42:PRINT"YOUR PROBLEM IS ";A;" DIVIDED BY ";B;
680 LOCATE 15,39:PRINT "      "
690 LOCATE 16,37:PRINT B$;"1";A$
700 LOCATE 16,48:PRINT MID$(A$,2,2);" ";CHR$(246);B;" =";
710 DX=INT(VAL(MID$(A$,2,2))/B)
720 INPUT DY
730 IF DX<>DY THEN BEEP:GOSUB 1880:LOCATE 16,48:PRINT STRING$(14," ");:GOTO 700
740 GOSUB 1880:LOCATE 14,41:PRINT DY;
750 LOCATE 17,47:PRINT " ";DY;"x";B;" =";
760 INPUT DM
770 IF DM<> DY*B THEN LOCATE 17,47:BEEP:PRINT STRING$(14," ");:GOTO 750
780 GOSUB 1880:IF DM<10 THEN LOCATE 17,41 ELSE LOCATE 17,40
790 PRINT DM:LINE(312,135)-(335,135),3
800 LOCATE 18,48:PRINT MID$(A$,2,2);" -";B*DX;"=";
810 DO=VAL(MID$(A$,2,2))-(B*DX)
820 INPUT DP
830 IF DP<> DO THEN LOCATE 18,48:BEEP:PRINT STRING$(14," ");:GOTO 800
840 LOCATE 18,41:PRINT DO;
850 LOCATE 18,43:PRINT MID$(A$,4,1);
860 DO$=STR$(DO):DV$=DO$+MID$(A$,4,1):DV=VAL(DV$):GOSUB 1880
870 IF DV<10 THEN LOCATE 19,48:PRINT DV;CHR$(246);B;" =";
880 IF DV>9 THEN LOCATE 19,47:PRINT DV;CHR$(246);B;" =";
890 DW=INT(INT(DV)/B)
900 INPUT DX:GOSUB 1880
910 IF DX<>DW THEN LOCATE 19,47:BEEP:PRINT STRING$(14," ");:GOTO 870
920 LOCATE 14,41:DY$=STR$(DY):DX$=STR$(DX):DZ$=DY$+RIGHT$(DX$,1):PRINT DZ$;
930 GOSUB 1880:LOCATE 20,48:PRINT DX;"*";B;" =";
940 DU=DX*B
950 INPUT DT
960 IF DT<>DU THEN LOCATE 20,48:BEEP:PRINT STRING$(14," ");:GOTO 930
970 IF DX*B<10 THEN LOCATE 19,42:PRINT DX*B; ELSE LOCATE 19,41:PRINT DX*B;
980 LINE(328,151)-(350,151),3
990 LOCATE 21,48:PRINT VAL(DV$);"-";(DX*B);" =";
1000 DV=VAL(DV$):DS=DX*B:RM=DV-DS
1010 INPUT RO
1020 IF RM <> RO THEN LOCATE 21,48:PRINT STRING$(14," ");:BEEP:GOTO 990
1030 LOCATE 20,42:PRINT RM;
1040 LOCATE 14,48:PRINT"REMAINDER =";RM
1050 LOCATE 22,45:PRINT"WRITE IN ";AD:GOSUB 1880
1060 LOCATE 8,10:PRINT"CORRECT! HOW ABOUT ANOTHER (Y/N)?"
1070 AK$=INKEY$:IF AK$=""THEN 1070
1080 IF AK$="Y"OR AK$="y" THEN 1100
1090 FOR WIPE=14 TO 22:LOCATE WIPE,37:PRINT STRING$(31," ");:NEXT WIPE:GOSUB 181
1100 FOR WIPE=14 TO 22:LOCATE WIPE,37:PRINT STRING$(31," ");:NEXT WIPE:GOSUB 181
1110 '
1120 ***** END PROGRAM *****
1130 '
1140 COLOR 3,0:LOCATE 16,42:PRINT"OK - NO HOMEWORK TONIGHT!"
1150 CIRCLE(350,145),25:CIRCLE(338,140),4:CIRCLE(362,140),4:DRAW"BM348,143;F2E2"

```



```

1160 DRAW"BM346,148;E2R4F2":DRAW"BM344,152;F2R8E2"
1170 LOCATE 19,48:PRINT"<- Teacher!"
1180 '
1190 GOTO 1190
1200 '**** THE CLASSROOM MAIN BODY ****
1210 '
1220 KEY OFF:SCREEN 6,1
1230 PALETTE 0,15:PALETTE 2,1
1240 PALETTE 1,2:PALETTE 3,4:CLS
1250 '
1260 '**** BRICK WALL ****
1270 '
1280 ' ENTERING THE BLACKBOARD!....ASCII Characters for the Blackboard are
1290 ' entered by holding down the ALT KEY and typing the ASCII number for
1300 ' the desired character. Characters used are as follows:
1310 '          205 is = / 203 is ¯ / 202 is ± / 206 is ÷
1320 '
1330 BK$="
1340 BM$="
1350 BD$="
1360 LOCATE 12,10:PRINT BK$
1370 FOR X=13 TO 23:LOCATE X,10:PRINT BM$:NEXT X
1380 LOCATE 24,10:PRINT BD$;
1390 '
1400 '**** THE WALL BOUNDARIES ****
1410 '
1420 LINE(75,190)-(0,200),1
1430 LINE(590,190)-(639,200),1
1440 LINE(75,90)-(0,85),1
1450 LINE(590,90)-(639,85),1
1460 LINE(0,80)-(639,80),1
1470 LINE(75,90)-(75,190),1:LINE(590,90)-(590,190),1
1480 PAINT (50,110),1,1:PAINT(600,110),1,1
1490 '
1500 '**** THE BLACKBOARD AND CLASSROOM DETAIL ****
1510 '
1520 FOR BOARD= 13 TO 22:LOCATE BOARD,35:PRINT STRING$(35," ");:NEXT
1530 LINE(274,95)-(550,176),2,B
1540 LINE(80,160)-(250,180),2,BF:LINE(80,180)-(90,190),2,BF:LINE(240,180)-(250,1
90),2,BF:LINE(84,162)-(246,178),0,B
1550 CIRCLE(100,150),12,2:PAINT(100,150),2,2
1560 LINE(96,154)-(98,160),3,BF:CIRCLE(98,148),3,1:PAINT(98,148),1,1:CIRCLE(102,
152),4,1:PAINT(102,152),1,1:LINE(225,150)-(230,160),3,BF
1570 LOCATE 19,29,0:PRINT CHR$(5);
1580 LINE(107,100)-(228,128),2,BF
1590 LINE(108,101)-(227,127),1,B
1600 LINE(107,124)-(136,120),1:LINE-(148,124),1:LINE-(170,119),1:LINE-(192,124),
1:LINE-(226,119),1:CIRCLE(140,111),3,1:PAINT(140,111),3,1:PAINT(109,126),1,1
1610 LINE(0,70)-(639,70),0:PAINT(5,72),4,0
1620 RC=1
1630 FOR RB=1 TO 6:CIRCLE(168,120),52+RB,RC,0,3.3,.3:RC=RC+1:IF RC=2 THEN RC=RC-
2:NEXT RB
1640 '
1650 '**** PUT UP THE TITLE ****
1660 '
1670 LOCATE 1,21:PRINT"***** The ELECTRIC BLACKBOARD *****"
1680 LOCATE 2,1:PRINT STRING$(80,"_");

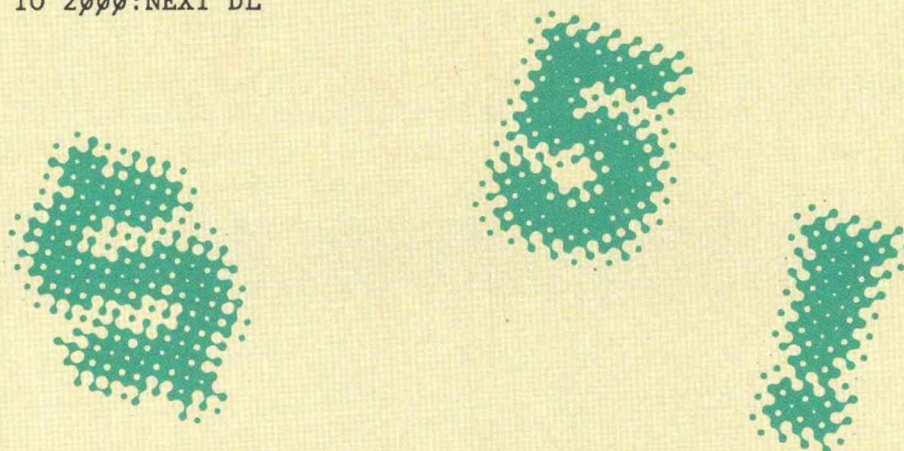
```

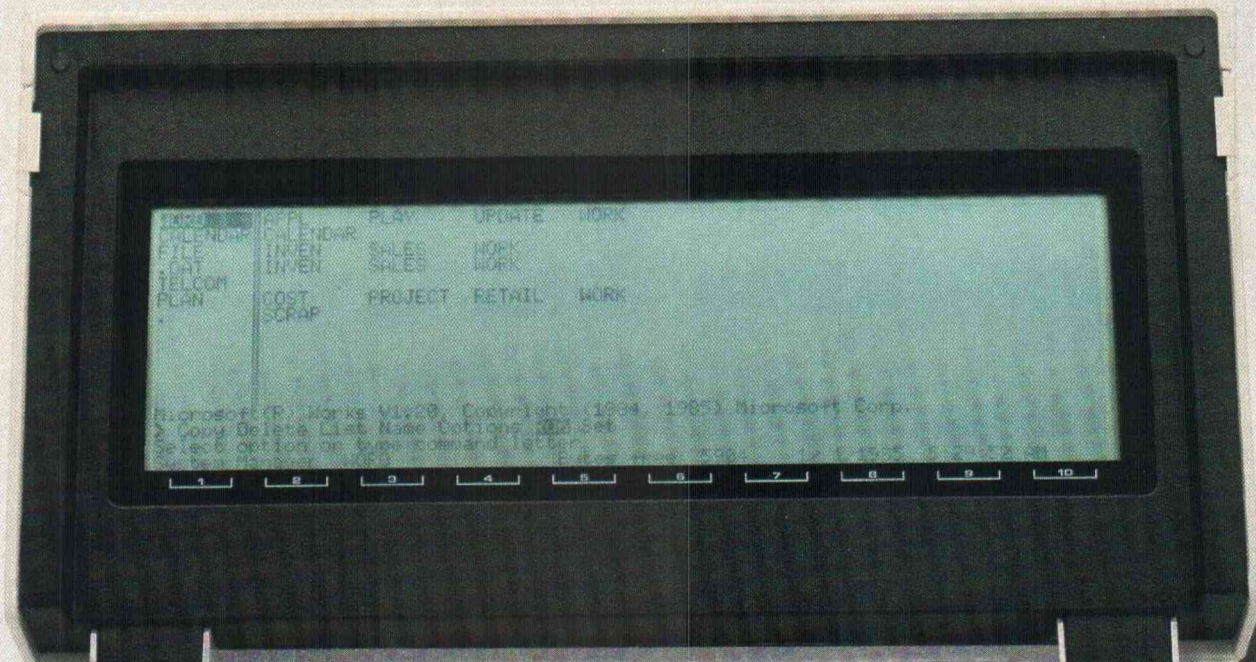



```

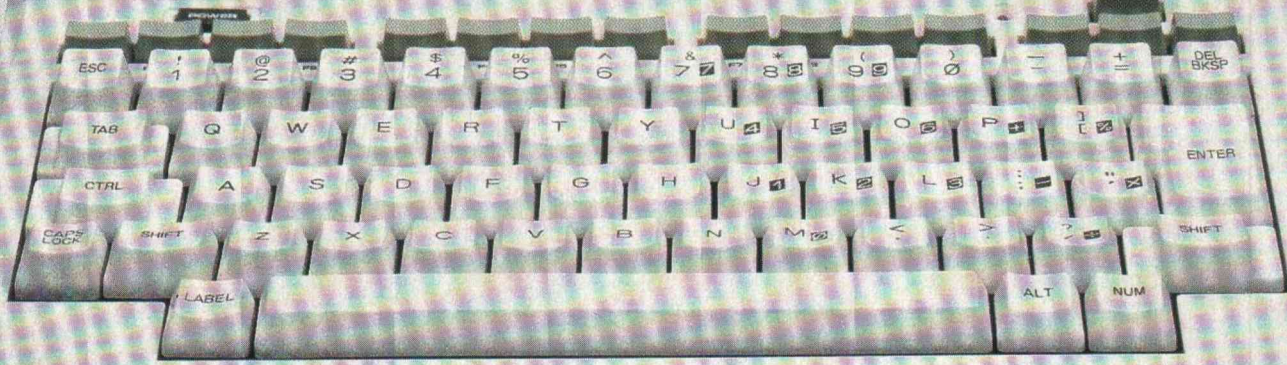
1690 RETURN
1700 '
1710 '**** THE PROGRAM MENU ****
1720 '
1730 LOCATE 4,36:PRINT"The MENU"
1740 LINE(10,20)-(630,70),2,B,&H5555
1750 LOCATE 5,35,0:PRINT"1> MULTIPLY":LOCATE 6,35:PRINT"2> DIVIDE";:LOCATE 7,35,
0:PRINT"3> SCHOOL'S OUT"
1760 AN$=INKEY$:IF AN$=""THEN 1750 ELSE IF VAL(AN$) <1 OR VAL(AN$) >3 THEN 1750
ELSE AN=VAL(AN$)
1770 RETURN
1780 '
1790 '**** WIPE OUT OLD SCREEN DISPLAYS ****
1800 '
1810 FOR WIPE=3 TO 9
1820 LOCATE WIPE,1:PRINT STRING$(80," ");:NEXT WIPE
1830 LINE(10,20)-(630,70),2,B,&H5555
1840 RETURN
1850 '
1860 '**** NOISE AS NEEDED ****
1870 '
1880 FOR DL=1 TO 300:NEXT DL:PLAY "V1503L64C":RETURN
1890 '
1900 '**** TITLE SCREEN ****
1910 '
1920 '[ SEE PCM 9/85-The Sounds of Science by Bill Barden FOR sound creation ]
1930 '
1940 KEY OFF:SCREEN 6:COLOR 12,0:CLS
1950 LOCATE 25,1:PRINT STRING$(80," ");:PALETTE 1,14:PALETTE 2,10
1960 LOCATE 12,28:PRINT"The ELECTRIC BLACKBOARD"
1970 LINE(50,40)-(590,150),2,B,&H5555
1980 LOCATE 22,35:PRINT "by L. Hyre"
1990 FOR LTNG=1 TO 6
2000 DRAW"BM250,110;C1G15R10G12R8G9":DRAW"BM375,110;C1G15R10G12R8G9"
2010 DRAW"BM200,95;C1G8H16G12H9":DRAW"BM200,95;C0G8H16G12H9"
2020 OUT &H61,&H6C:OUT &HC0,&HE0+1*4+0:FOR I=1 TO 15:OUT &HC0,&HF0+I:NEXT I
2030 DRAW"BM250,110;C0G15R10G12R8G9":DRAW"BM375,110;C0G15R10G12R8G9"
2040 DRAW"BM250,80;C1H15R12H12R8H12":DRAW"BM375,80;C1E15L12E12L8E12"
2050 DRAW"BM250,80;C0H15R12H12R8H12":DRAW"BM375,80;C0E15L12E12L8E12"
2060 OUT &H61,&H6C:OUT &HC0,&HE0+1*4+0:FOR I=1 TO 15:OUT &HC0,&HF0+I:NEXT I
2070 DRAW"BM410,95;C1F12E18F8E12":DRAW"BM410,95;C0F12E18F8E12"
2080 NEXT LTNG
2090 FOR DL=1 TO 2000:NEXT DL
2100 RETURN

```

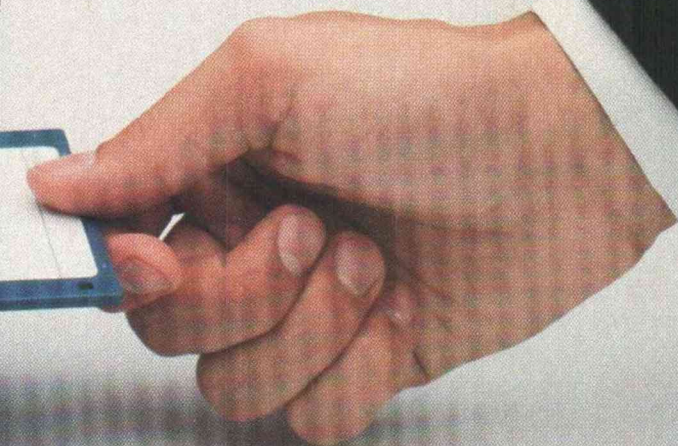




TANDY 600 PORTABLE COMPUTER



The new disk-based Tandy 600.



A breakthrough in portable computing power and versatility.

Finally—a portable that matches the performance of a desktop computer. Our new Tandy 600 features a 16-bit microprocessor, an 80-character by 16-line display, a built-in 3¹/₂" disk drive that stores 360K of data and 32K RAM (expandable to 224K*).

Five Resident Applications

With the Tandy 600's larger display and expanded memory, Multiplan spreadsheet analysis can hold more information. Word processing is as easy as using MS-Word. You'll have quicker access to documents, and better storage with the built-in disk drive. File is an electronic data base for names and addresses, expenses, client billing, inventory and more. And you can keep a large number of different files on the pocket-size diskettes. With Telecom and

the Tandy 600's built-in modem, you're able to communicate with other computers over phone lines and access national information networks. Telecom will even dial the phone number of anyone listed in the File program. Calendar helps you keep track of daily tasks and activities.

The easy-to-learn resident System Manager lets you run each application, exchange information between applications and manage the files created. And you can add BASIC/ROM (26-3904, \$129.95) to write your own programs.

Tandy . . . Clearly Superior™

If you've been looking at portable computers, compare them with the Tandy 600 (26-3901, \$1599). You'll be amazed at what it can do!

Available at over 1200
Radio Shack Computer Centers and at
participating Radio Shack stores and dealers.

Radio Shack
COMPUTER CENTERS
A DIVISION OF TANDY CORPORATION

Send Me a Free Tandy 600 Brochure.

Radio Shack, Dept. 86-A-916
300 One Tandy Center
Fort Worth, TX 76102

Name
Company

Address
City
State
Zip
Phone

* RAM upgrades are in banks of 96K each (26-3910, \$399.95). Prices apply at Radio Shack Computer Centers and participating stores and dealers. Multiplan and MS-Word/TM Microsoft Corp.

Going Online with Telecom: Computerize your Logon Procedure with Analog Files

By Bobby Ballard

Last month we were discussing many of the aspects of telecommunicating, specifically the Telecom section of *DeskMate*. This particular aspect of *DeskMate* has many features and can be confusing, especially if you are new to telecommunicating. This month I want to cover the features we didn't get to last month so you can make complete use of Telecom. I won't normally cover the same subject two issues in a row unless the topic, as in this case, is rather involved and lengthy. In addition, I feel it is important for you to put the Telecom section to work so you can reach out and get quick and efficient help from PCM by accessing the MSDOS SIG on Delphi.

In the last installment I didn't have the time or space to cover "autolog" files other than to give a glossary explanation of what they are. We did cover auto-dialing, status settings and other aspects of telecommunicating which are directly related to using the autolog feature of Telecom. By using the auto-dial and autolog features you can easily connect with any other system with very little effort on your part.

What is an autolog file? Simply stated, it is a predicted conversation between two computers, host and terminal, that you keep stored on disk to use in making connection. By knowing what information the computer you are calling is going to request, you can design an autolog file that handles the information exchange automatically.

What is the difference between autolog and auto-dial files? The autolog file described above uses your auto-dial file to handle the actual dialing of the phone number, through the modem, and then uses the rest of the autolog file to handle the actual information exchange after connection is completed. You only design one auto-dial file for computer dialing that is designed to handle your modem. If you buy a new or different modem with a different protocol then you will have to redesign this file.

(Bobby Ballard is a free-lance writer and the owner of a computer software and consulting firm. He also operates a BBS in Brooklyn. Bobby can be contacted at 1207 Eighth Avenue, Apt. 4R, Brooklyn, NY 11215.)

By comparison, you will design and save many autolog files, which handle each of the services or BBSs you might be calling on a regular basis.

It is nearly impossible to design an autolog file without knowing something about the service you will be calling. Since you are designing a predicted conversation, you must know exactly what is coming in order to provide the answers to each of the prompts. For example, if you are designing an autolog file for logging onto Delphi, you must not only know that the host system, Delphi, is going to ask for your password, but you need to know exactly how the system will ask for the password and when. Will the prompt be "Enter your password:" or will it be "PASSWORD:" or even something different?

In order to know what is coming, it is often necessary to manually logon to the service you will be using and keep track of the information exchange and prompts as you go through them. This is easy to do since you can just logon with your buffer open. This will cause all of the information to be stored in your computer memory; from there it can be saved to diskette after you log off for use in building your autolog file.

Once you have saved your buffer to diskette, make a printout of the information you have and you're ready to start designing your autolog file. Make sure you have designed your auto-dial file and tested it so that you can concentrate on getting the autolog portion working correctly. If you know that the auto-dialing feature is working correctly and run into problems with an autolog procedure, then you will know where the problem is *not*.

You will notice that each autolog file you design shows up in your Telecom window on *DeskMate's* main menu screen. This makes it possible to enter Telecom by invoking an autolog file and immediately begin logging on. Just make sure your modem is switched on and ready. Then move the file highlighter in the Telecom window to the autolog file that matches the service you will be calling and press ENTER. *DeskMate* then enters Telecom, loads your autolog and auto-dial information and begins to make the connection.

I want to point out one very distinct difference you will notice when saving autolog files on your diskettes. Unlike

other files, autolog files are stored on the system diskette regardless of whether you are using the swap feature. In other words, if you are using Drive B: to save and retrieve information from Text or Filer and then design an autolog file; the autolog file will be saved to the system (*DeskMate*) diskette. Likewise, when you swap drives from A: to B: your autolog files will still appear in the Telecom window. There might be a way around this but I've not discovered it yet.

Take a look at Listing 1 for an example of the type of information that would appear in your printout after manually logging onto the Delphi information utility. The only difference would be the username and password information. This example is from logging on using the Tymnet packet switching network. If you use Uninet or Datapac, then the listing would be different, but the procedure we are going to follow would be the same.

Listing 1

```
please type your terminal identifier A
-4515-006-
please log in: DELPHI
host: call connected
Username: YOUR USERNAME
Password: YOURPASSWORD
```

To get started building your autolog file for Delphi, first enter the Telecom section of *DeskMate* and then press F4 for Editlog. You will then be greeted with an autolog edit screen for editing your file. Now think of the procedure you went through when logging onto Delphi, using the printout as an aid, and begin building.

Notice the set of commands on the bottom of the autolog edit screen represented by F1-F5 and two operations represented by F9 and F10. The first five function keys are not direct commands, but commands that you place into your file. Using the function keys above each command you are able to enter these functions into your file with very little typing. Think of these keys as macros. Instead of having to type the word CALL, for example, you just type F1 and the word appears on the screen followed by a cursor.

The first step in logging onto any service is to set your terminal status. This comprises the different protocol settings found on the status screen that we covered last month. In order to save the specific settings needed for each of your autolog files you use the F1 key for Status. After pressing the F1 key, you will see the Status screen where you may change the settings to whatever is desired before saving them in your autolog file. Once set, press F12 (may be different on some machines) and you will be back in autolog edit.

If the above status setting procedure is successful, you will see something similar to the first line in Listing 2. Each of the numbers has a meaning that relates directly to the status settings you chose. The first number is the Baud rate (shortened) followed by the number of bits (7 or 8), parity (letter 'N', 'E' or 'O'), the number of stop bits, the answers to the next four settings (On or Off) and the number of retries the autolog file is to use.

The last setting is directly related to the autolog file itself and not to the communications status to be used while online. This is the number of times an autolog file will attempt to logon before giving up and aborting the procedure. By entering a zero in this parameter, the autolog file will try only once with zero retries. Entering a '1' will

Listing 2

```
STATUS: Y,30,B,N,1,ON,OFF,OFF,OFF,0
CALL: 12129434700
PAUSE: 2
RECV: terminal identifier
SEND: A
RECV: please log in
SEND: DELPHI^M
RECV: Username:
SEND: YOUR USERNAME^M
RECV: Password:
SEND: YOURPASSWORD^M
```

try to logon twice, once for the first time and then one retry. So the total number of times the autolog will try to logon is one more than the number of retries because you must add the first time to the number of retries.

The next step is to tell the autolog file what number to dial for making the connection. To do this you use the Call feature and enter it into your file by typing F2. The word CALL will appear on the screen followed by the cursor waiting for you to input a number for dialing. At this point type in the phone number you use in your local area for accessing Delphi. Type it in using just numbers; no spaces, no dashes. Check Listing 2 for an example.

The next feature to invoke is the PAUSE. This is a way of telling the autolog file to wait a specific length of time, in seconds, before proceeding to the next command. You may have to insert a pause at various locations in your file to match the time it takes for various systems to respond to your data. In this case we are giving the modem and the host two seconds to get their signals together before we start sending data from the remaining autolog file.

Most of the time the first two settings (lines 1 and 2 in Listing 2) will remain the same in all of your autolog files. Each of the status settings may be different for different services, but the status command will almost always be the first line in your autolog file. You will also almost always have the third line (PAUSE) in all of your files. However, the amount will vary.

The next step in setting up your file is to begin referring to the printout you made from your manual logon. You must remember to send and receive the information that the network requires, as well. In my example, using Tymnet, the first request is "please type your terminal identifier." I have shortened this request to the last two words. This works fine and saves labor.

One very important point to remember is to always construct the RECV: information exactly like it appears in your printout. Uppercase and lowercase are treated differently, and it is especially important to type in the exact number of trailing spaces that are found in the host's prompts. In Listing 1 you will notice that a space appears between the colon and the answer I've provided. This space is sent by the host for neat formatting and must be included in your RECV: design.

The next line in the file sends an 'A' without a carriage return. The next send, though, must be followed by a carriage return represented by ^M. This is called a control character and in ASCII code the carriage return is the 'M' preceded by a caret. In the manuals that came with my copy of *DeskMate* there is some confusion concerning how to

place a control character in your autolog file. The manual explains that you hold the control key CTRL down followed by the character 'M'. This will not work, though. You must use the shifted '6' key to place the caret symbol on the screen and into your file.

Some systems will require that you send a BREAK signal to initiate communications. For example, CompuServe requires a BREAK signal. To send the BREAK signal you represent it with the caret symbol (SHIFT 6) and then a capital 'C'. To figure out which letter will send which control character, just subtract 64 from the ASCII code number of that letter. The carriage return ASCII code, represented by decimal 13, is 64 less than the ASCII code for 'M' which is decimal 77. So, to send a carriage return (CR) just enter a caret (^) followed by a capital 'M'. This is a carriage return or control M (CTRL-M).

You will be using the control M extensively while creating your autolog files. When you send a line of text, username or password, it must be followed by the ENTER key when logging on manually. To duplicate this action in your autolog files you must send the control 'M' which is represented by the caret symbol followed by a capital 'M' as explained above. Just remember to use the SHIFT 6 to achieve a caret symbol instead of actually hitting the control CTRL key.

Following my example in Listing 2, you should now be able to create your own autolog file for logging onto Delphi and accessing the MS-DOS group where you can leave me or the PCM staff a message. If you are still having problems, then logon manually and leave me a message describing your question or problem and I will try to track down the answer for you.

To end and save your new autolog file, you first press the F12 key, after which you will see the filename prompt displayed at the bottom of the screen along with the name with which you first entered the Editlog section. If this name is still the one you wish to use, just hit the ENTER key and your file will be saved. If you wish, you may change the name of the file before striking the ENTER key. The extension doesn't need to be provided in all cases, but if you do, it must be .LOG in order for *DeskMate* to recognize it as an autolog file.

This ability to change the name of your autolog file before saving it back out again allows you to use pieces of other files without always having to start at the beginning with each new file. You can load any autolog file, change the appropriate information and then change the name before hitting the ENTER key which will then create a new autolog file.

Now that you have a file created, you will want to put it to work. You may invoke an autolog file in two ways. The first and easiest way is to select the autolog file as you are entering the Telecom section of *DeskMate*. This is similar to selecting a specific text file before entering the Text section of *DeskMate*. You just place the highlighter bar over the name of the file and press ENTER. You will be taken to the Telecom section and it will automatically begin to log you onto the service you selected.

The second way to invoke an autolog file is from the Status screen in Telecom. To do this you use the F3 key and supply a filename for Telecom to use. When you press F3 you will again see a prompt for a filename at the bottom of the screen and, if you already have an autolog file loaded, the existing filename will also be shown. If this is the file you wish to use, just press ENTER. Of course, if you wish

to use another file then type in the file name and press ENTER. You do not need to include an extension but must use .LOG if you do.

That's it! If your modem is on and you have done everything correctly, the computer and *DeskMate* will take it from there. As the file is doing its logon, you will be able to see the information on screen just as if you were typing it in by hand. This is also a good way to spot problems in your autolog files. You are able to see exactly where the problem is and how to change it if necessary.

To help you get an idea of some of the different autolog files you might create, I've included two other listings for two additional autolog files. Listing 3 is for logging onto CompuServe which you may use if you have an account there. Just replace the name and password information with the specific information from your account. The final listing, Listing 4, is for logging onto MCI Mail. Again, if you have an account, use this with the appropriate information inserted. If you don't have accounts at either of these services, just use them as a reference when creating other autolog files.

One final note. Remember, you can extend your autolog files to take you directly to a section you wish to spend time in. For example, if you wish, you may add a couple of lines like the following to your Delphi file and have it take you directly to the MSDOS group (the hyphen is omitted from MS-DOS in Delphi usage) before turning control back over to you.

```
RECV: you want to do?  
SEND: GROUP MSDOS^M
```

You can, of course, extend this even further to take you to the forum and begin reading messages. You might even consider having these different versions on your disk directory under different names depending on what you wish to do on each call. One file might be called DELPHI, the next might be called DELMSDOS and so on. I'll leave it up to you, at this point, to experiment and explore the autolog section of *DeskMate*. I hope you have fun and will stop to say hello to all of us on the MSDOS SIG. □

Listing 3

```
STATUS: Y,30,8,N,1,ON,OFF,OFF,OFF,0  
CALL: 12127584114  
PAUSE: 5  
SEND: ^C  
RECV: User ID:  
SEND: 73135,255  
RECV: Password:  
SEND: Your password
```

Listing 4

```
STATUS: Y,30,8,N,1,ON,OFF,OFF,OFF,3  
CALL: 12122450355  
PAUSE: 5  
SEND: ^M  
RECV: your user name:  
SEND: USERNAME  
RECV: Password:  
SEND: YOURPASSWORD
```

PCM

*BASIC programming error recognition
made easier*

Easy Errors

By John and Aileen Cornman

In the interest of conserving memory, BASIC error messages are the soul of brevity. Until the two-letter codes are committed to memory, all but the ubiquitous SN error probably need to be looked up each time they occur. The experienced user overcomes this first difficulty only to encounter a second. Now committed to packing as many commands into a line as the law will allow, the problem becomes one of recognizing *where*, rather than *what*, the error is.

Memory is indeed a scarce resource, especially on, say, an 8K Model 100. But to us it seems that the user should have a choice about how to "spend" that resource. Since both the novice and the veteran have their own reasons for desiring more help in finding programming errors, we offer an alternative BASIC error handler that will cost you between 778 and 1,562 bytes (depending on how you choose to set it up).

Listing 1 installs a machine language error handling program in high memory which performs two major services. First, it displays a longer, more descriptive form of the error message. Second, it displays the line containing the error and, whenever possible, helps you locate where within the line the error was detected. These two functions are performed automatically and in place of BASIC's normal error message whenever an error occurs.

Our alternative error handler can be switched on and off at will by means of two POKE commands. Because it is written in machine language it does not

interfere in any way with ON ERROR routines which may be included in the BASIC program which is being debugged.

Installing the Program

Before running Listing 1, you must use the built-in TEXT program to copy Listing 3 into your Model 100 as HEX.DD. Listing 1 reads and verifies the HEX.DD file while loading the corresponding instructions into an area of protected memory. It checks for invalid characters in the file, and whether the total of all the characters agrees with the pre-determined "checksum" total. If no errors are found, ERRORS.CO is created and is loaded into its run-time memory location.

No harm will be done if you try to run ERRORS.CO from the main menu because it is stored with an "entry point address" that only causes an immediate return to the menu.

To conserve memory, you may delete the loader program, the HEX.DD file and ERRORS.CO (after backing them up on tape, of course). Once ERRORS has been loaded into memory and properly protected, it can function without the files that were used to create it.

You may continue to use ERRORS, even though all visible signs of it have been deleted, as long as you do not run a program or enter a command which removes its protection (such as CLEAR 100, MAXRAM). Once left unprotected, you should *not* try to use the ERRORS program until it has been reloaded and re-protected or you might end up having to "cold start" the machine, losing everything in RAM.

By leaving ERRORS.CO as a file on the menu, you can easily check whether its run-time memory area is still protected. If you try to run ERRORS.CO from the main menu when its memory area is no

longer protected, the Model 100 beeps before it redisplay the menu. If still protected, there is no beep as the screen goes blank then returns to the menu display.

Alternatively, if you have deleted ERRORS.CO from the menu, you may enter the command, PRINT HIMEM. If the value returned is 62,181, you can be reasonably sure the run-time copy of ERRORS is still intact and safe to use. If these precautions are not taken, using the ERRORS program could produce unpredictable results.

Using ERRORS

Enter BASIC and load the program you want to debug. Add the following lines to the very beginning of your program:

```
1 POKE 63058,230
2 POKE 63059,242
```

These two commands activate the ERRORS program in place of BASIC's usual error routine. You may also accomplish the same result by entering the same commands in the immediate mode. In either case, the ERRORS program is automatically "switched off" whenever you return to the main menu.

Once activated, ERRORS reports on errors instead of BASIC. For example, if you run a program containing the line

```
20 IF A=0 THEN B=B$+C ELSE B=0
```

ERRORS displays

```
Type mismatch
20 IF A=0 THEN B=B$+C ELSE B=0
```

The underlined portion of the line is displayed in the reverse character mode (white letters on black background). The location of the error coincides with

(Aileen and John Cornman live in Battle Ground, Wash., where they own and operate their own computer business, Structured Software Services.)

the right edge of the string of reversed characters. The actual cause of the error may be either to the immediate left or to the immediate right of the edge.

In the example above, the shading would divide the line into two parts, with the B\$ (which is the cause of the error) being just to the left of the dividing line. You can imagine the reversed characters to be like a guage that starts at the left of the line and

extends to the right until the error is encountered.

ERRORS always prints the long error message and always displays the line containing the error, but it is not always possible to determine the exact location of the error. Since BASIC does not have a consistent way of keeping track of error locations, all that can be done is to check three of its most popular hiding places for the information. As a result,

ERRORS may sometimes provide only rough guidance, and occasionally may not be able to give any location information at all.

Summary

The ERRORS program provides the beginner and experienced programmer alike with significantly more information about the cause and location of BASIC errors than is given by the standard ROM-based error routine. □

Listing 1:

```

2  ***          ERRORS Loader          ***
4  *** Structured Software Services ***
6  *** 9233 N.E. 269th Street ***
8  *** Battle Ground, WA 98604 ***
9  '
10 CLEAR 100,62181:CLS
12 DEFINT A-Z:DEFSNG M:DEFDBL C:I=0:CS=0
14 M=62182
16 OPEN "HEX" FOR INPUT AS 1:PRINT
18 PRINT"Loading HEX.DO into memory..."
20 HX$=INPUT$(2,1)
22 D$=LEFT$(HX$,1)
24 GOSUB 82
26 IF D=-1 THEN 68
28 DC=D*16
30 D$=RIGHT$(HX$,1)
32 GOSUB 82
34 IF D=-1 THEN 68
36 DC=DC+D:CS=CS+DC
38 POKE(M+I),DC:I=I+1
40 IF EOF(1) THEN CLOSE ELSE 20
42 IF CS=85610 AND I=778 THEN 54
44 PRINT:PRINT"Checksum error."
46 PRINT"Please re-check HEX.DO file"
48 PRINT"and rerun Loader with the"
50 PRINT"corrected file."
52 END
54 PRINT
56 PRINT"Successful load."
58 PRINT
60 PRINT"'ERRORS.CO' will be created."
62 PRINT
64 SAVEM"ERRORS",62182,62959,32755
66 END
68 CLOSE:PRINT
70 PRINT"Non-hex character in HEX.DO:"
72 PRINT
74 PRINT"pair #";I+1;"reads: ";HX$
76 PRINT
78 PRINT"Please correct and rerun."
80 END
82 D=ASC(D$)
84 IF D>47 AND D<58 THEN D=D-48:RETURN
86 IF D>64 AND D<71 THEN D=D-55:RETURN
88 D=-1:RETURN

```

Listing 2:

```

2A7AF6229FFB7CA53CCA00F322A1FB2A8EFB22F0
F3E1E522EEF30109F32A9DFBC3783FC17B4B3272
F62A9BFB22A3FBEB2A9FFB7CA53CCA2CF322Aafb
EB22ACFB2AA5FB7CB5EB21A7FBCA3BF3A6C23BF3
35EBC32B08AF7759CDB84B7BFE3BD251F3FE32D2
53F3FE17DA56F33E30D61B5F21F2F3AF1DCA67F3
23BEC25EF323C35
AF3CDB127CD2242A9FFB7CA53CCA0205EBCD280
6E503030303030303030303030303030303030
7C1EB2AEFF3DFDA9FF3D55059DFD1DABDF32AF0F
3DFDAAEF3D55059DFD1DABDF32A9BFBDFDAE4F3D
55059DFD1D2E4F37EF53600EBCDAA11F12B7703C
DB911CD69422185F6CDA211E5CD6E42E123CDA21
1CDCB4BC30205EB
CDA112185F6C3DBF300000000000000000000000
74686F757420464F520053796E74617820657272
6F720052455455524E20776974686F757420474F
535542004F7574206F66206461746100496C6C65
67616C2066756E6374696F6E2063616C6C004F76
6572666C6F77004F7574206F66206D656D6F7279
00556E646566696
E6564206C696E650042616420737562736372697
07400446F75626C792064696D656E73696F6E656
4206172726179004469766973696F6E206279207
A65726F00496C6C6567616C20646972656374005
4797065206D69736D61746368004F7574206F662
0737472696E6720737061636500537472696E672
0746F6F206C6F6E
6700537472696E6720666F726D756C6120746F6F
20636F6D706C65780043616E6E6F7420636F6E74
696E756500492F4F206572726F72004E6F205245
53554D4500524553554D4520776974686F757420
6572726F7200556E646566696E6564206572726F
72004D697373696E67206F706572616E6400496E
7465726E616C206
572726F72004261642066696C65206E756D62657
20046696C65206E6F7420666F756E6400416C726
5616479206F70656E00496E70757420706173742
0656E64206F662066696C65004261642066696C6
5206E616D65004469726563742073746174656D6
56E7420696E2066696C65004469726563746F727
92066756C6C0046
696C65206E6F74206F70656E00

```

PCM

Adding a hard disk and memory — a guide for the do-it-yourselfer

Upgrade Your Tandy 1000

By Brian Alsop

The Tandy 1000 offers a low-priced alternative to the IBM PC. However, one quickly finds out that it often needs some extras. Two frequently required extras are more RAM and a hard disk. This article describes how to add these two options to your Tandy 1000.

Memory expansion comes by way of a PBJ MFB-1000. This expansion board also adds a serial port and real-time clock.

The hard disk added is a 20-megabyte QUBIE PC-20. At \$599, it is a cost-effective way to add a hard disk. The drive also comes with some useful software for managing files on the hard disk. The initial concerns about using third-party hardware proved to be unfounded. They both work well together and with the Tandy 1000.

PBJ Memory Expansion Board

PBJ'S MFB-1000 offers a serial port, real-time clock, direct memory access (DMA) circuitry and memory expansion of up to 512K bytes. The board appears to be well constructed, uses quality circuit board material and has gold-plated contacts. The fact that it occupies only a single slot and is less expensive than equivalent Tandy boards is also a plus!

Installation is as easy as opening the computer and plugging in the board. Installation of the software to access the clock is also quite easy.

(Brian Alsop is an engineer working on introducing personal computers into the engineering work environment. He is a pilot and holds an amateur radio operator's license.)

Installing the Memory Board

Unplug the computer, loosen the two front panel screws, slide off the case, remove one expansion slot cover plate and plug the PBJ board into that slot — any slot can be used. PBJ recommends precautions be taken against static electricity in this process. Now, attach the single screw to the PBJ cover plate to secure the board inside the computer.

The manual explains where to set PBJ board switches if less than 512K of memory is installed. Mine came with the switches properly set. Memory is now so inexpensive that it doesn't make any sense to get less than 512K with the board. Slide on the computer case and secure the screws.

Testing the Memory Board

Turn on the computer and the amount of available memory should immediately appear. If it is less than the sum of the PBJ board and the computer's 128K memory (640K with a 512K expansion board), immediately turn off the computer and look for switch setting errors on the PBJ board. If the proper memory is displayed, a memory check will be automatically performed. If no errors are found, the computer continues the process of booting up and displays an A> prompt.

Now, one is ready to test and install the clock software. Two steps are necessary. The first requires adding a CON FIG.SYS file to the MS-DOS disk. If you don't already have such a file, copy PBJ's from their disk. If you already have such a file, add a line saying `DEVICE = CLOCK.COM` to it with `EDLIN`.

Next, copy the file `CLOCK.COM` to the MS-DOS disk. Reset the system (press `CRTL-ALT-DEL` simultaneously or press the red reset button). At the `A>` prompt, type `TIME` and press `ENTER`. The system will display the time. If it is correct, press `ENTER`, otherwise enter the correct time. Now, type `DATE` and press `ENTER`. Correct the date if necessary. Subsequent resets of the system will not require any date or time entries. The clock date and time information is automatically included in all file saves and can be accessed from `BASIC` and other languages as well as most commercial programs. The unit comes pre-tested and it is unlikely that any problems will be encountered. However, PBJ suggests that problems first be handled via the supplier. They also give a phone number to use to discuss technical problems not resolved with the supplier.

There are some drawbacks to the PBJ board. First, it must be used as communications Port 1, `COM1`. Any other communications expansion board must be configured as `COM2`. The second problem lies not with the PBJ board but with the Radio Shack-supplied software.

On IBM equipment, the communications port can be configured as a printer port with `MODE LPT1:=COM1,9600,,,P`. The last parameter, `P`, is required to prevent the port "timing out" between characters or lines. The Tandy `MODE` command doesn't allow such a parameter. As a matter of interest, the `IBM MODE.COM` program will run on the Tandy 1000.

To date, I've had no difficulties with

any software accessing the extra memory. Even virtual disk software works just fine. An expansion board of this nature turned out to be necessary to utilize BASIC graphics screen modes 5 and 6.

Hard Disk Installation

Installation of the Qubie PC-20 internal mount hard disk is a little more complicated than installation of the PBJ memory expansion board. I didn't consider less than a 20-megabyte drive because of my previous experiences with IBM PCs. I had a PC XT and filled up the 10-megabyte hard drive in less than six months. With a 20-megabyte drive, I've found that I quickly used 12 megabytes. The storage use then increased slowly with time.

The commercial programs that one purchases uses up most of the storage. Once you've collected the programs to do your job, the remaining storage space decreases more slowly. The instructions supplied with my version didn't address installation in a Tandy 1000 — only other IBM PC compatibles. Needless to say, it required a few calls to Qubie's technical assistance line to accomplish the task.

Qubie does supply everything that is necessary to install the unit in a Tandy 1000. The assistance line people are quite helpful and understanding.

The installation involves three steps:

- 1) Installation of the controller board,
- 2) installation of the hard disk and
- 3) formatting. Before doing anything, make sure the Tandy 1000 is unplugged. Don't even think about working on the unit with the power cord plugged in.

Inspect the controller card. Locate the EPROM chip. It should have a sticker on it. Make sure the label on the sticker says Tandy 1000 on it somewhere. If it doesn't, the controller board is for an IBM PC or compatible. Call the factory and arrange to get the correct EPROM. This problem can be avoided by specifying in the order that the PC-20 is to be used for a Tandy 1000.

Before installing the controller card, one must open the Tandy 1000. There are two screws on the front panel of the Tandy 1000. These must be loosened. Remove the cabinet by sliding it forward. Opening the Tandy 1000 does not void the warranty, so you need not wait until the warranty expires to do this installation. Once removed, locate a free expansion slot. The controller ribbon cables are long enough to reach any slot. I chose the second slot next to the PBJ board.

This is when you will find out the

TANDY™

TANDY 2000



SEE WHAT

Someday, our competition

TANDY computers are created equal. . .all retailers are not.

A good price is, obviously, very important but there are "other important differences" which the buyer should be aware of, before parting with any money. A good example is "the retailers buy-back policy*", to insure the buyer against that expensive mistake, the call free number for easy price comparison and credit card ordering for convenience and safety. The price quoted should be the total price with no road-freight charges (to be added later). Under normal circumstances the order should be shipped the next business day and a "same day rush-service" should be available.

* For a small restocking fee.

**CUSTOMER SERVICE/QUESTIONS ABOUT
YOUR ORDER and in TEXAS 1-817-573-4111**
(9 am-5 pm TEXAS TIME MONDAY-FRIDAY)

Fort Worth Computers
377 Plaza
Granbury, Texas 76048



CALLFREE NUMBER:

COMPUTERS and PRINTERS etc.

TANDY 1200

WE OFFER.
will give as much.

The fact that the majority of orders are from "repeat-buyers" is a strong indication of the great satisfaction and trust that our customers feel. However, for those still not convinced we offer "references" to prove our bona fides and "that the Radio Shack warranty and service will be available, to our customers, NATIONWIDE.

These eight important considerations will be yours when you order from Fort Worth Computers. Call (free) for a price comparison.



FORT WORTH COMPUTERS

(WE ARE SERIOUS ABOUT SAVING YOU MONEY)

Located 30 miles from Ft. Worth

For
Latest
Prices

(1-800-) **433-S-A-V-E**

controller card doesn't just plug in — the metal mounting bracket is for an IBM and must be sawed off at the bottom to fit inside the Tandy 1000. Remove the blank mounting bracket covering the intended slot. Use it as a template to cut the Qubie mounting bracket. If you wish to remove the Qubie mounting bracket from the controller board, it is attached with only two screws. Pay attention to how the bracket is mounted so that re-installation will be correct. Mark where to cut using the Tandy 1000 blank as a template and make a straight-across cut. Don't bother trying to make it have a tongue like the Tandy 1000 bracket. Re-attach the bracket on the controller board.

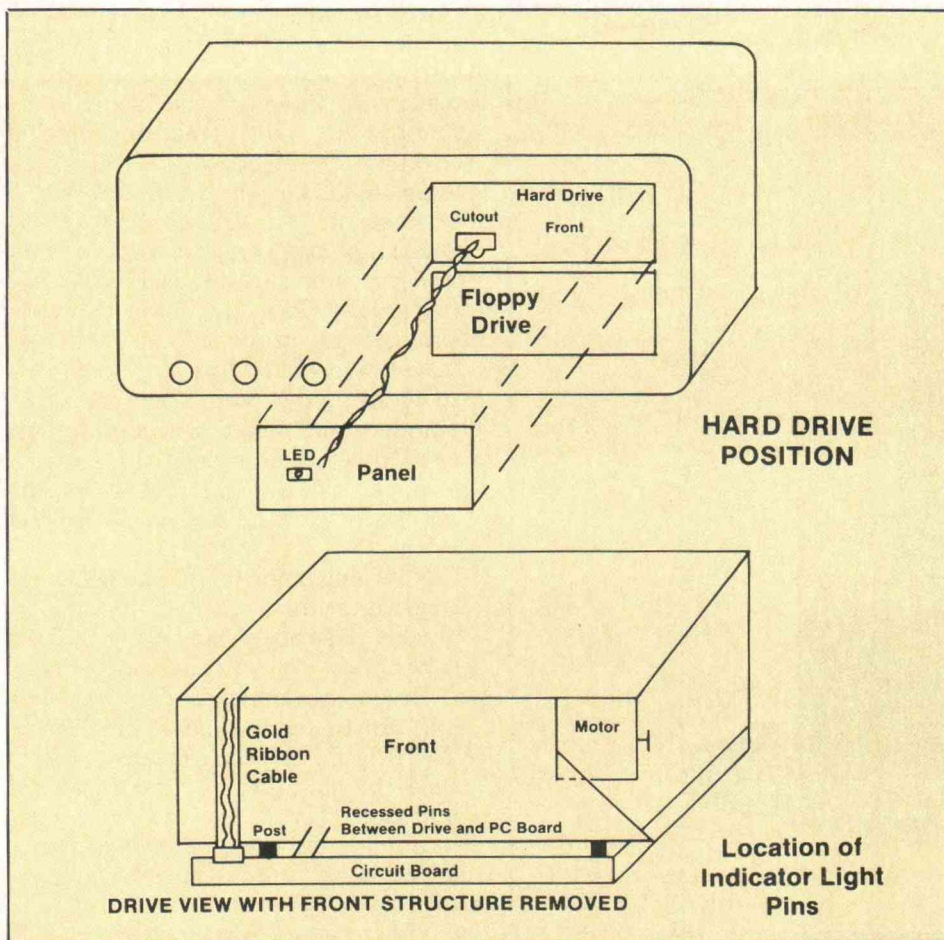
Now plug in the board and fasten the single attachment screw to secure the bracket. This board can only be plugged in one way — the correct way.

Route the cables under the metal partition to the area in back of the drive. There is barely enough clearance to slide the connectors under the metal partition. Try various front or back locations until the connector makes it under the partition. Route the large 34-wire cable first.

The supplied instructions refer to switch settings necessary for the controller and hard disk to work properly. These instructions are for the IBM PC — ignore them.

Drive preparation and installation requires a little improvisation on the Tandy 1000. There are two front panel bezels supplied with the hard disk. The smaller half-height bezel is used here. The instructions tell you to plug the indicator light into the two corresponding pins on the disk drive printed circuit board. The picture supplied gives one no hint of where they are. Their location is shown in Figure 1. The connector is not keyed and can be inserted incorrectly, however, incorrect installation of the plug does no harm. The indicator is an LED and won't light if the plug is installed incorrectly. In my unit, the plug had a "P7" tag glued to one side of the plug. The correct installation in my case was with the tag side facing away from the PC board. Route the wire through the drive front panel cutout provided and push the plug onto the two pins. Now snap on the front cover and make sure all snaps fully seat. Route the two ribbon cable plug ends so they hang out the front of the computer where the drive is to be installed. The drive goes above the floppy drive.

Ignore the rest of the supplied instructions in the drive preparation



section. They don't work for a Tandy 1000. Instead, cut a piece of cardboard from a box slightly larger than the drive's printed circuit board. Place this piece on the metal shelf above the floppy disk drive.

Push the two ribbon cable connectors on the back of the hard drive. They are keyed and can only be inserted correctly. Slide in the hard drive and make sure the cardboard spacer doesn't shift. Now connect the four-prong power connector into the female socket. The matching socket is inside the Tandy 1000 near the back of the floppy drive. This connector is also keyed by shape and can't be inserted improperly.

Now mount the hard drive to the Tandy 1000 with the two provided screws. These two screws both go on the right side of the drive. If you have some dexterity, a magnetic screwdriver and an extra screw that fits, put a third screw into the left-hand side of the drive. This is a difficult job to do since it requires working through a slot in the drive enclosure cabinet. You have to use feel instead of sight.

Dress the drive ribbon cables neatly and replace the cover but don't tighten the screws — you might have installed the indicator light plug backwards! The



BARBAROSSA

The War in Russia 1941-44

ARK ROYAL'S most popular Color Computer wargame is now available on the Tandy 1000 and all IBM compatibles. Hi-res graphics, 100% ML, Barbarossa is the same game raved about in the RAINBOW and HOT COCO (Jan '86), except that the 1000's memory allows room for a host of expanded features. Included are bitmapped logical characters (tanks, infantry, airflottes, cities and terrain), normal and Blitzkrieg movement, airstrikes, supply consideration, unit detail, group transfer, weather, intelligence, (which even tells you which sector your attacking unit is in), patrols, game save and even more. \$24.95, disk. We pay shipping and handling on all prepaid orders in USA. COD charge is \$2.50. Personal checks in US accepted with no shipping delays. Foreign orders send M.O. in US funds. FL residents add 5% sales tax.

Program requires disk, graphics card, 128K.

P. O. Box 14806
Jacksonville, FL 32238
 (904) 786-8603

hard drive installation is now complete.

Drive initialization cannot be done on a Tandy 1000 with the instructions or software supplied. Read the directions in the instructions anyway to get an idea of what has to be done.

There are three steps. The first formats the disk with a Qubie program, QUBIEFMT. The second step is to partition the disk. In practice, the disk could be partitioned into separate independent areas to accommodate different operating systems. Finally, format the disk again and copy the "hidden" operating system files to the disk. This permits booting and startup from the hard disk by itself.

The second and third steps require the Tandy 1000 MS-DOS programs FDISK and HFORMAT. PC-DOS versions will not work with the Tandy 1000 ROM installed on the hard disk controller card. My MS-DOS diskette did not contain either program. You have to obtain them from Radio Shack. I managed to get them free by going to a local Radio Shack Computer Center and talking to them. It doesn't hurt if you order something (like the *Tandy 1000 Technical Reference Manual*) or buy something at the same time. While you're there get the latest copy of MS-DOS and BASIC. Versions generated later than June 15, 1985 include a number of fixes to BASIC and some very useful MS-DOS enhancements. Once all the programs are available, follow these steps.

Insert the MS-DOS disk in the floppy drive, close the door and turn on the computer. The machine will go through

its startup ritual.

Insert the Qubie-supplied diskette and type QUBIEFMT. The program will load and the hard drive light should light. This formatting process takes over five minutes. If the hard disk indicator light did not light, you probably installed the indicator plug backwards or forgot to plug in the disk drive power cable. Listen to see if the drive is operating. The program will complete with a message indicating successful formatting.

Re-insert the MS-DOS diskette into the floppy drive and re-boot the system (CTRL-ALT-DEL keys simultaneously depressed).

After startup, insert a diskette with the FDISK, HFORMAT and MS-DOS programs into the floppy disk drive. Type FDISK and press ENTER. The FDISK program will present a menu.

Select "Create DOS Partition" and divide the disk into one partition.

Next, select the "Change Active Partition" option and activate Partition 1. You must activate the partition to permit booting from the hard disk. This step to activate the partition is a quirk of the Tandy 1000/MS-DOS version of FDISK. It is not necessary on IBM PCs.

Select the option to "Display Partition Data."

You should see that the disk has been divided into only one piece with 611 cylinders allocated for the partition. The status table should indicate that the partition status is active (A).

Exit FDISK with the ESC key and type HFORMAT C:/S/V and press ENTER. This program will format the disk and copy

the hidden and COMMAND.COM file to the hard disk.

You will be prompted for an 11-character "volume name." Enter any name up to 11 characters.

If any step along the way fails, try going back to QUBIEFMT and repeat the process. If it fails again call Qubie.

Now, turn off the machine, open the floppy disk drive door and turn the machine on again. The startup ritual will begin. This time the unit will go to the hard disk for MS-DOS and command files. It speeds up the process considerably.

If you forgot to activate the partition with FDISK, the unit will not boot up from the hard disk and issue a "no active partition" error. Go back to FDISK and activate the partition. It is unnecessary to reformat the disk.

To see what you have done by all this, type CHKDSK at the next prompt. This program will return the number of bytes of storage available. In my case it was 21,225,472. All tracks were good. Don't expect this to always be the case. Even IBM equipment only guarantees 20 megabytes to be good on their system — even though the disk usually has more. The extra bytes are a way of guaranteeing a high production yield of units.

If all the above processes work properly, tighten the screws on the front cabinet.

Qubie supplies a program called 1DIR with their software. This program provides a user interface to the hard disk and permits easy creation of files, file directories and BAT files.

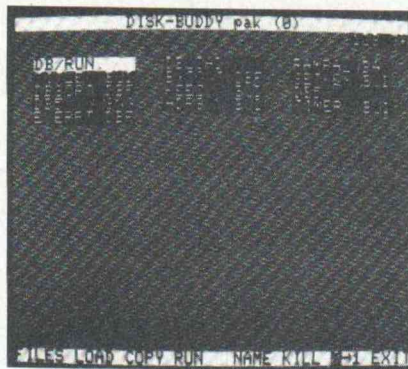
Civilize the DISK/VIDEO INTERFACE • MAKE SAVAGE THE MODEL 100

*With **DISK-BUDDY** software for your DV/I, only function keys need touch your disk files!*

Twelve programs for one low price, work together to give the TRS-80 Model 100 unique disk file handling abilities. Access disk files from menu, just like a RAM file. Commands you were tediously typing become simple function key selections. And that's just part of the power of the DISK-BUDDY PAK. It does things you can't:

- Create "superfiles" the length of the disk, over 170,000 bytes!
- Print out, or display on either screen, or copy to disk or other device, a disk data file of any length, **WITHOUT USING ANY ADDITIONAL RAM!**
- Furthermore, transfer records can be "filtered", an example:

An address file on disk can contain more than 1300 entries. You can move to RAM, or screen, or printer, etc., those



records which contain "California" and "Tennis" or whatever. Four logical modes of search are available.

- Also: Features to run, load, save, sort, kill, measure, append, and name, disk files and RAM files, and much more.

PCM Magazine (March, '85) said: "... an excellent buy" ... "manual is extremely well written" ... "will substantially increase the speed and ease of operation of the M-100 with the D/VI."



That was the idea in the first place.

TO ORDER THE DISK-BUDDY PAK (disk and manual): Send name, address, and check or money order for \$39.95 plus \$2.00 postage and handling to the address below. (For COD's call: 212-243-2129)

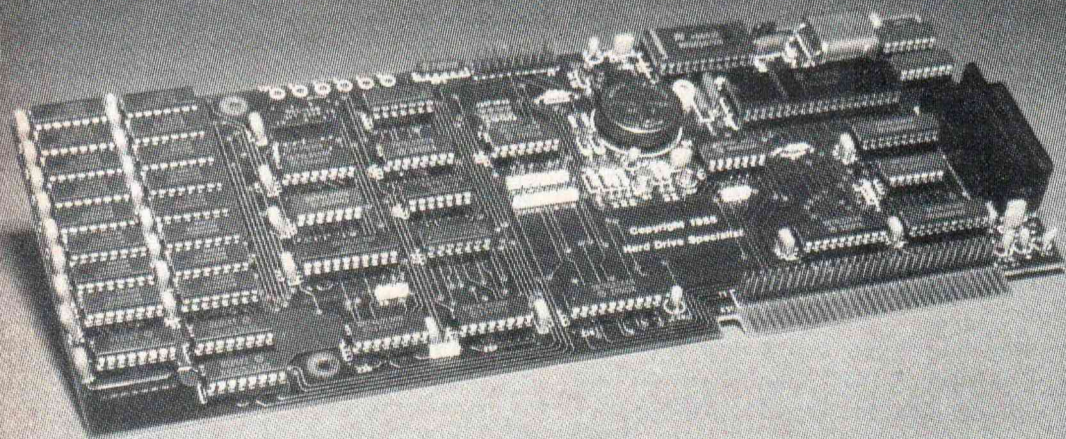


BuddySystems

220 West 24th Street
New York, NY 10011

The Ultimate Expansion for the Model 1000!

TanPak™ by Hard Drive Specialist

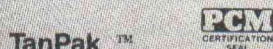


7 Functions Include:

- DMA
- Serial Port
- Memory up to 512K
- Clock/Calendar
- Printer Spooler
- Memory Disk
- Expansion Port for Future Options

Features Include:

- Gold Edge Cards
- Pre-Tested and Burned In
- Full Documentation
- Supporting Software
- One Year Warranty
- Expandable to 512K



TanPak™

The TanPak™ expansion board has been designed to allow expansion beyond the scope of the standard Model 1000. Seven of the most needed functions/features have been combined into one package using only one expansion slot. Your remaining spaces are left free for future expansion needs.

Seven Function/Features on One Board

512K Memory Expansion

Socketed and expandable to 512K. This is done by two banks of memory using either two 64K increments (128K), or 256K increments (256K or 512K). This allows a total of 640K in the Tandy 1000.

Serial Port

Using the same configuration as the Model 1000 port you are assured of complete compatibility as well as being able to configure it as COM1 or COM2.

Clock-Calendar

Quartz-controlled for a high degree of accuracy, featuring a battery backup.

DMA

The DMA (Direct Memory Access) is used on the Model 1000's first memory card. It increases memory speed and is a must for hard drive operation.

Printer Spooler

Use part of your TanPak™ memory as a printer buffer. Choose the amount of buffer space you need and stop waiting on your printer.

Memory Disk

Use part of your TanPak™ memory as a RAM drive. With a solid state drive you can store, retrieve, and sort data quickly and easily.

Expansion System

The TanPak™ was designed with an expansion port that will allow upgrading with additional features when they become available. Some of the possible features are: a second parallel port, a second serial port, mouse, hard drive port, bubble memory, A/D, and D/A as well as many others.

TanPak™ 0K.....\$329.

Call for pricing on memory modules.

TanPak™ Secondary

If you already have a Model 1000 memory board and do not wish to replace it, the TanPak™ Secondary is for you. It features Memory up to 256K, Clock/Calendar, Serial Port, Printer Spooler, and Printer Buffer.

TanPak™ Secondary 0K.....\$249.

TanTel™

Internal 300/1200 baud Modem for the Tandy 1000. Supported by the DeskMate software package.

TanTel™.....\$249.

Ordering Information

Use our W.A.T.S. line to place your order via Visa, MasterCard, or Wire Transfer. Or mail your payment directly to us. Any non-certified funds will be held until proper clearance is made. COD orders are accepted as well as purchase orders from government agencies. UPS ground is our standard means of shipping unless otherwise specified. Shipping costs are available upon request. Tandy and Radio Shack are trademarks of the Tandy Corporation. TanPak, and TanTel are trademarks of Hard Drive Specialist, a division of Compukit Corporation.

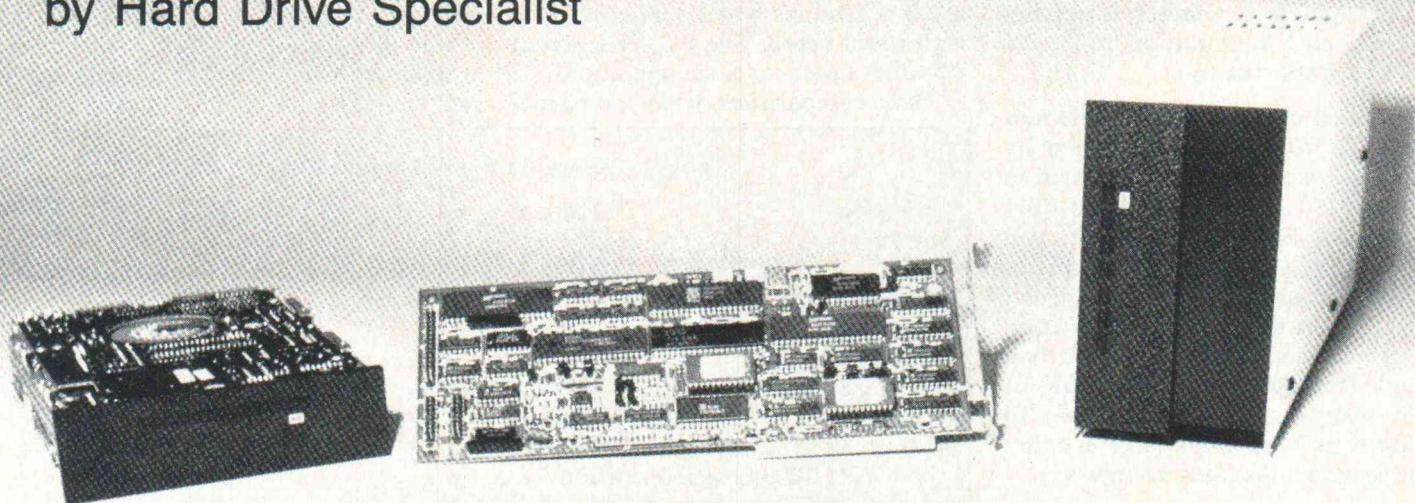
Hard Drive Specialist

16208 Hickory Knoll
Houston, Texas 77059
1-713-480-6000

Order Line
1-800-231-6671

Megabytes for Mini-Bucks

Tandy 1000 Hard Drives
by Hard Drive Specialist



Shop and compare. Hard Drive Specialist has been building hard drive systems for years and have sold thousands of subsystems to satisfied Radio Shack/Tandy customers. We use the latest state-of-the-art disk drives and controllers. Our drives all use buffered seek logic and plated media to result in almost one-fourth the average access times found on our competitors drives. Plated media results in longer platter life and high resistance to head crashes not found on coated media drives. We utilize high quality construction throughout including heavy duty power supplies and gold edge card connectors. Internal drive systems include an interface card and a half-height hard drive that replaces the top disk drive in both size and power consumption. External units include an interface card, case, power supply and hard drive unit. All units require a memory board with DMA such as the TanPak™ multifunction card, or a Tandy Board, Part #25-1004.

Hard Disk System Features:

- : All sizes rated after formatting.
- : One secondary drive may be added.
- : Error checking and correcting controller.
- : Software drivers included.
- : Buffered seek drives for improved access time.
- : Built in power up diagnostics.
- : Plated media for long disk life.
- : Gold connectors used throughout.
- : 1 year warranty.
- : Boots directly from Hard Drive.
- : Uses Tandy 1000 MS DOS.

10 Megabyte	Internal	\$ 549.
20 Megabyte	Internal	\$ 749.
42 Megabyte	Internal	\$1349.
10 Megabyte	External	\$ 699.
20 Megabyte	External	\$ 899.
42 Megabyte	External	\$1495.

Ordering Information

Use our WATS line to place your order via Visa, MasterCard, or Wire Transfer. Or, mail your payment directly to us. Any non-certified funds will be held until proper clearance is made. COD orders are accepted as well as purchase orders from government agencies. UPS ground is our standard means of shipping unless otherwise specified. Shipping costs are available upon request. Tandy and Radio Shack are trademarks of the Tandy Corporation. TanPak, is a trademark of Hard Drive Specialist, a division of Compukit Corporation.

Hard Drive Specialist

16208 Hickory Knoll
Houston, Texas 77059
1-713-480-6000

Order Line
1-800-231-6671

Because of a hard disk's enormous volume, it is useful to divide its contents up into separate directories. Think of a directory as a drawer in a file cabinet. Put your files into directories like you would put documents in a file drawer. Group them logically, such as word processors and associated files in one directory, BASIC programs in another, etc. While programs in one directory can access those in another, it requires a special path designation to be included with the filename.

One word of caution: With the new BASICA fix to GW-BASIC supplied by Tandy, the usual PATH C:\ inserted in the AUTOEXEC.BAT file on startup, does not find BASIC.EXE. Therefore, you must put BASIC.EXE in the directory in which BASIC programs are executed. I hope Tandy re-writes their BASICA.COM file to eliminate this glitch. It's bad enough that BASIC occupies 70K of RAM instead of being in ROM. To require it to be in every directory that BASIC is used adds insult to injury.

Speed Comparisons

To satisfy my curiosity, I did a number of speed comparisons between the upgraded Tandy 1000 and other

computers. The computers compared included the IBM PC XT, IBM PC AT and the Tandy Color Computer. BASIC was used exclusively. Table 1 shows the results.

As one can see, there is precious little difference between the Tandy 1000 and PC XT. The Color Computer is also very close to matching the Tandy 1000 and PC XT, and is even faster for some math operations when operating in its high speed mode. The PC AT is about 2.5 times faster for most operations.

These comparisons, while not partic-

ularly rigorous, indicate that you don't replace a Color Computer with a Tandy 1000 or PC XT just for speed. Note the math operations are done in BASIC and do not use the 8087 or 80287 math co-processor chips. These chips, when added to the PC XT and PC AT with appropriate software, speed up math operations five to 20 times. The one thing missing from the Tandy 1000 is the ability to install such a co-processor. I would be eternally grateful to someone who writes an article describing how to add an 8087 to the Tandy 1000. □

Table 1. Computer Speed Comparisons

Operation	T-1000	CoCo	PC-XT	PC-AT
Subtract	41	44/33	38	15
Add	41	44/31	38	15
Multiply	43	68/50	40	15
Divide	43	75/53	49	19
I/O Floppy	27	23/—	25	14
I/O Hard Disk	19	—	20	8

Notes: 1) Times are in seconds.
2) 10000 math operations
3) I/O problem READ/ WRITE of a file containing 1000 lines of "HI THERE"
4) First number in CoCo column is normal speed mode, second is high speed.

PCM

Discover What 10 Software Insiders Already Know!

1. "... a little company that has succeeded over the last five years in giving its customers a lot of value for their money ..." — Richard O'Reilly, computer columnist
2. "All the firm's programs are graced with features that the entire industry should be required to provide. Their manuals are the most honest in the industry." — T.R. Reid, syndicated computer columnist
3. "... the two most important attributes of Software Toolworks' programs: they're priced low, and they work well." — The Washington Post
4. "Unlike the software of many publishers these days, none of The Software Toolworks programs are hobbled by copy protection, nor does the company try to tie up its customers with mumbo-jumbo licensing-agreements ..." — The Los Angeles Times "Computer File"
5. "I don't know how The Software Toolworks stays in business charging such low prices for such nifty products ..." — David Gerrold, computer columnist and noted science fiction author
6. The Software Toolworks C/80 compiler is "... the best software buy in America ..." — Microsystems
7. "It would be the best of all possible worlds if some magic diskette could be made that would transform any hardware in such a way as to receive The Software Toolworks' numinous software." — Wm. F. Buckley, Jr.
8. "The Software Toolworks has a whole raft of programs that work, and (the company) sells them for reasonable prices. I strongly recommend that you get their catalog." — Dr. Jerry Pournelle, computer columnist "Byte" and noted science fiction author
9. "One of the original pioneers in low cost, high quality software, The Software Toolworks publishes over 50 solid products in MS-DOS and CP/M for under \$60." — Walt Bilofsky, chairman of The Software Toolworks
10. "All products are shipped within 24 hours, come with a money back guarantee, and free technical support is available (though you'll hardly need it)." — Joseph Abrams, president of The Software Toolworks

Reading Professor — Boost your reading speed while increasing comprehension and retention. Self paces lessons show quick improvement with immediate feedback. Includes 48 practice reading passages and other exercises. Developed by skilled educators, this program tracks your progress and monitors reading rates up to 2,500 wpm. Additional libraries available for Junior High, College Adult, and Professional levels at \$15.95 each. For IBM PC, most CP/M-80 computers. — \$59.95.

TO ORDER CALL: 1-800-223-8665; in CA 1-800-228-8665

The Software Toolworks



14478 Glorietta Drive, Sherman Oaks, CA 91423

U.S. and Canada add \$2.00 per order s/h. Overseas orders add \$5.00 per order. CA residents add 6.5% sales tax. Orders shipped UPS. Dealer inquiries invited.

Mychess — Award-winning world class chess program, featuring nine skill levels and over 850 opening moves. Prints moves and saves games in progress. Outstanding graphics display board on IBM PC and most popular CP/M computers. — \$34.95

Eliza — The pioneering artificial intelligence program. Eliza is a "psychiatrist" that carries on a conversation in plain English. Use authentic script, expand it or develop your own. CP/M and MS-DOS. — \$24.95

The Original Adventure — Explore the dangerous Colossal Cave for treasures. Endure mystical spells, overcome adversaries. The only official version, uncut, and endorsed by its creators, Crowther and Woods. CP/M and MS-DOS — \$19.95

Computer Chef — Original computerized cookbook and recipe file. Data base of over 70 recipes with space to add your own. Scale ingredient quantities, search recipes by keyword, then display or print your selection. Additional recipes available. CP/M and MS-DOS. — \$29.95

TANDY IS DANDY...

...until you want more memory

Your Tandy 1000 is a good machine.
A lot of computer for the money.

But you need to add more memory to
get the best performance possible.
Tandy wants a lot of money for that.
Too much money.

But, now there's another way to upgrade
your memory. The Zuckerboard
Expansion Memory card. A high quality,
reliable board at a very reasonable price.

So, if you want to get the most out of
your computer, you have a choice.

You can go to Tandy, and Tandy will get
the most out of you.

Or, you can come to us. We'll fill up your
computer without emptying your wallet.

Board Configuration	Tandy Memory Plus	Zuckerboard Expansion Memory
with 0K	N/A	\$ 69
with 256K	\$319.95	\$109
with 512K	\$519.90	\$149
Clock Option	\$ 99.95*	\$ 39
Warranty	90 Days	2 Years
	*with mouse	

MADE
IN
U.S.A.

CALL TOLL-FREE TO ORDER

(800) 624-4920
CA (800) 458-6200

Dr. Dr. Zucker



It's another
ZUCKERBOARD



1287 Lawrence Station Road • Sunnyvale, CA 94089 • (408) 734-4631

ZUCKERBOARD is a Trademark of Advanced Transducer Devices, Inc.
Tandy 1000/1200 are Trademarks of Radio Shack, a Division of Tandy Corporation.



WE'RE BRINGING THE TANDY

PCM's BROADENING ITS SPECTRUM

PCM and the Delphi Information Utility have joined together to allow Tandy MS-DOS owners all over the world to connect with one another!

Delphi is a full-service information utility. It offers everything from up-to-the-minute news stories from The Associated Press to electronic mail services. But, *best of all*, it now has a **special forum for owners of MS-DOS computers, and it's operated by the people who bring you PCM each month.**

The MS-DOS Special Interest Group (SIG) features a variety of services, including an open forum where you can send and receive messages from Tandy owners all over the world. It also has databases for exchanging programs, where you can upload your favorites and download programs written by other Tandy MS-DOS enthusiasts.

When setting up your account with Delphi, if you do not have a credit card or prefer not to use it, Delphi requires that you send \$20 to give your account a positive balance. This will be refunded after your first free hour if you choose to no longer use the system or it will be applied to future connect charges.

THE WORLD OF MS-DOS

The MS-DOS SIG's conference feature allows you to meet electronically with other members of the Tandy Community. You can join conferences with many notables in the world of MS-DOS on a regular basis. Conference schedules will appear in PCM each month. Be sure to check online announcements for changes and additions.

PCM ONLINE

On Delphi, You can renew your PCM subscription, make a fast and easy order for software or hardware from a multitude of vendors, or inquire about products on the MS-DOS SIG.

We also have a number of **programs that you can download** and use, just for the cost of the time you spend transferring them. There'll also be **corrections for PCM articles**, helpful hints and many other useful features.

FREE LIFETIME MEMBERSHIP

PCM is offering subscribers a free lifetime subscription to Delphi — a \$29.95 value — and a free hour of connect time — a \$6 value at either 300 or 1200 Baud during evening, holiday and weekend hours — so you can sample Delphi and the brand new PCM MS-DOS SIG. **That's right. Your subscription to PCM entitles you to this \$35.95 value as a free bonus!**

If you're not a PCM subscriber, just enter your order when you sign on with Delphi and you'll get the same great deal! For our \$28 subscription fee, you'll get the finest Tandy computer magazine ever, a free lifetime subscription to Delphi and a free hour of connect time.

SAVE EVEN MORE

Want to save an extra \$15? While you're online, you can order the *Delphi Handbook and Command Card* (\$21.95) and three hours of connect time (\$18) for only \$24.95.

Delphi provides us all with **Immediate Tandy Community.** Check it out today. After all, you can sample it for free!

Problems? Call Delphi at 1-800-544-4005.

DELPHI

TYPE: GROUP MSDOS



COMMUNITY TOGETHER!

How to reach the MS-DOS SIG on Delphi . . .

There are several ways to connect to Delphi and PCM's MS-DOS SIG. In most cities you will not even have to pay long distance charges; you can use special data communications networks, like Uninet, Tymnet and the Canadian Datapac network.

First, set your terminal program to operate at either 300 or 1200 Baud (depending on the modem you have), and also select either 7 bits with even parity or 8 bits with no parity, and 1 stop bit. (If one combination doesn't work, try another.)

Decide which network you should use. There is no surcharge for Uninet or Tymnet. Canadian residents using Datapac will be charged an additional \$3 (U.S.) per hour.

On Uninet: Call (800) 821-5340 to get the Uninet number for your area. After you call the appropriate number for your own area and make connection, press ENTER, the period key (.) and ENTER again. At the "service:" prompt, type GVC (for General Videotex Corporation) and ENTER.

On Tymnet: Call (800) 336-0149 to get the Tymnet number for your area. After you dial your designated number and make connection, simply press 'A' no matter what appears on the screen. When "please log in:" appears, type DELPHI and press ENTER.

From Canada (on Datapac): Call Delphi Customer Service at (617) 491-3393 to get the Datapac number for your area. After you connect, press the period key (.) and ENTER (use two periods if you're using 1200 Baud). Type SET 2:1, 3:126 and press ENTER. Now type p 1 3106, DELPHI; and press ENTER. Delphi will bill you an additional \$3 (U.S.) per hour for Datapac connection surcharges.

From other countries: Many countries have their own data networks that can connect to either Uninet or Tymnet. Check with the telephone authorities in your country for details on how to sign up for this service. When you have an account set up, you can reach Delphi with a "host code" of 312561703088 through Uninet, or 310600601500 through Tymnet. (You'll have to pay the toll charges for this connection.)

Type in Your User Name

If you're already a subscriber to PCM, at the "USERNAME:" prompt, type PCMSUB and press ENTER. At the "PASSWORD:" prompt, type your individual sub-

scription number from the mailing label of your latest issue of PCM. (If there are one or more zeros at the beginning of this number, include them.)

If you don't already have a subscription, at the "USERNAME:" prompt, type PCMORDER and press ENTER. At the "PASSWORD:" prompt, type SENDSUB and press ENTER. Have your MasterCard, VISA or American Express card ready, because you'll be led through a series of questions that will enable us to put your PCM and Delphi subscriptions into effect. In an effort to hold down non-editorial costs, we do not bill for subscriptions.

If you make a typing error, just press ENTER and start over. Remember that at any point, when you're on Delphi, you can type HELP to get help on how to use the system. To get off the system just type BYE.

If you find that you're unable to log onto Delphi and enter the MS-DOS SIG after following these instructions, call us during afternoon business hours at (502) 228-4492. We'll be glad to offer assistance.

Come Visit Us! Type: GROUP MSDOS

After you sign in, you'll be prompted to set up your own, personal "user name" — Delphi is a friendly service, no numbers to remember — and you'll be asked a number of questions so Delphi can set up your account. You'll also be assigned a temporary password. No time is assessed against your free hour of service while you answer these questions.

Delphi will tell you that your account will be ready after 6 p.m. the same day if you sign up before noon (Eastern time zone.) If not, your account will be ready at 6 p.m. the next day. Once an account is opened, *each PCM subscriber will be credited with an hour of free time!*

When you log back in, use your chosen user name and your temporary password to access the system. At that point, you will meet Max, who will help you configure things and will change your temporary password into your own *personal* password. This is the password you will use for subsequent sessions — or until you change it. Again, there is *no charge* for the time you spend configuring your account. Your hour of free time starts after Max bids you goodbye.

You'll wind up at the Delphi Main Menu; type in GROUP MSDOS and **join us on the MS-DOS SIG!**

The Wind Chill Factor

By Wayne Sanders

It's a typical February morning in Chicago. The thermometer by the kitchen window tells you the outside temperature is a normal 35 degrees, yet when you step out the door, dressed for 35-degree weather, the brisk winds off the lake make it feel like 10 below. This effect, of course, is called "wind chill" — something you've no doubt heard mentioned on the 5 o'clock news weather report many times.

If you know the outside temperature and wind speed, you can use this program to help you know how to dress and what to expect when you leave your warm home behind — if you even want to leave home.

This BASIC program adapted from a program by Stanley Kendall will run on the Tandy 1000, 1200, 2000, 3000, 100 and 200 (just about any computer that uses Microsoft BASIC). If you have a Tandy 600 with the optional BASIC ROM installed, you will also be able to use the program. □



The listing:

```

10 CLS
20 PRINT ">>> Windchill Calculator >>>"
30 PRINT
40 LINE INPUT "Temperature (F): ";T$:T=
  VAL(T$)
50 LINE INPUT "Wind speed (MPH): ";V$:V=
  VAL(V$)
60 IF V<4 THEN W=T:GOTO 100
70 B=10.45+6.686*SQR(V)-.44*V
80 C=33-5*(T-32)/9:D=33-B*C/22.03405
90 W=D*9/5+32
100 PRINT USING"Windchill:      ###.##"
    ;W
110 PRINT
120 LINE INPUT "Another? (Y/N) ";A$
130 A$=LEFT$(A$,1):IF A$="Y" OR A$="y" T
  HEN RUN
  
```

PCM

It's the Holmes Engineering/PCSG "chipmunk"

PORTABLE

Disk Drive

AS FAST AS RAM

**ASK ABOUT
BUDGET MODEL
\$399.**

PC only

for the Model 100/200

**NEW LOW PRICE
\$499.**

**Includes a library of
six powerful programs**



Uses the main menu concept. You see the disk directory instantly, arranged on your M-100 screen like your main menu. Just move the widebar cursor and transfer files with a function key. You can run a file directly from the diskette with the ENTER key. Uses 3½" microfloppy diskettes that have a rigid plastic casing and a metal core. They're tough and nonflexible. You can carry several in a shirt pocket without damage. There's 358K on a diskette. Ten of these in your briefcase and you've got 3½ megabytes.

Drive weighs only three lbs. and it works directly from the 110 outlet and recharges at the same time. It recharges in six hours with thousands of pages transferred between charges. It's compact, with dimensions of 2¼" x 5½" x 7.5"; and fits easily into your briefcase along with your Model 100 or 200.

Machine code programs, BASIC programs, *Lucid* files and documents all are saved and retrieved with no protocol—instantly, ready to run.

In a special association, Holmes Engineering and PCSG have worked together combining the hardware knowledge of Holmes and the software expertise of PCSG. The result is a product that can only be regarded as excellent.

**You see the disk directory
instantly; works just like the
main menu**

Here is what is really exciting. The portable disk drive has Random Access. Included as part of the operating system in the drive (ROM) is a very powerful disk BASIC.

This means that you can have BASIC programs that will access the diskette and read and write records directly on the diskette.

Just imagine yourself with this kind of capability.

Database—The portable disk drive stores your mailing list, inventory items, part #s and descriptions or any other data that you need to recall.

358K on a diskette

Invoice (purchase order)—At the touch of a button you can print out your sequentially numbered, professionally done invoices. This is truly professional invoicing capability.

Purchase orders are just as easy.

Sort—This excellent utility allows you rapid sorting of any records you have compiled. You can write the newly sorted list back in the same file on the diskette or to a new file.

Telecom interface—If you are a user that likes to access other computers or databases (for example CompuServe) by telephone then this powerful facility alone is worth the price of the disk drive. You can automatically download and upload information onto the diskette.

Calendar—Everyone who has seen this program has said, "This is the first calendar/

diary/scheduler on any computer anywhere that I can use. It is so functional."

The calendar program is usable for two reasons, first it is designed correctly, and second you have the memory (358K) on the diskette to log and access a tremendous amount of notes over a long period of time.

Personal Finance Manager—This wonderful program truly lets you keep track of your finances.

All your records are kept on the diskette. Bank accounts (checking and savings) and charge accounts such as MasterCard and Visa.

We at PCSG believe we have the ultimate Model 100 system, the Portable Disk Drive plus the *Lucid* spreadsheet on snap-in ROM, *Write ROM* word processing and the new 64K RAM expansion now available from PCSG.

We want you to find out for yourself at no risk. If you aren't totally satisfied within 30 days, simply return the disk drive for a full refund. Priced at \$499.00, including the software library. MasterCard, Visa, COD.



PORTABLE
COMPUTER
SUPPORT
GROUP

1-214-351-0564

11035 Harry Hines Blvd., #207
Dallas, Texas 75229

PCSG offers a 30-day/money-back trial!

Delphi Bureau

By Kevin Nickols
MS-DOS SIG Manager

Welcome to "Delphi Bureau," a new column where we will keep watch on PCM's newest effort to get information to our readers about the Tandy computers. The Delphi MS-DOS SIG is a new group and we still have a lot to learn about it ourselves. But we have more than 1,200 members already and are getting more and more services installed. The MS-DOS SIG will be an important extension of the magazine to us all as we learn to use and enjoy our computers.

Delphi is designed to be self-explanatory. When you first sign on, you will find it to be "menu intensive." This means that at each prompt, everything you can do is precisely displayed for you on the screen in a menu in plain English. This is known as "prompt level three." It makes learning the commands very easy. Once you have become familiar with the commands, you can switch to one of two other prompt levels. Level two (`/P 2`) does not display a menu, but lists the options on the prompt line. Level one (`/P 1`) is for the experienced users. It gives only the prompt, and greatly increases the speed at which you move around in Delphi.

Another feature that saves time is the ability to use abbreviations. To initiate any command, you just have to enter the fewest number of letters that makes your choice clear. For example, if you reach the MS-DOS SIG from the main menu, you can just type `G MS` instead of `GROUP MSDOS`. Most of the time only one letter is required. When more than one command begins with the same letter, you will need to use two or three letters.

At almost every prompt, typing `HELP` brings up a detailed help file with the available options. In some cases, typing a question mark (?) will serve the same purpose. You may also type `HELP OPTION`, where `OPTION` is the function you want help with, and receive data on that particular subject.

Better still, when you need help, I and many other people who help put PCM together will be there online to answer your questions. We may be contacted easily by leaving a message in the group Forum area, through the electronic mail service, and can even be paged and called into Conference for a little personal conversation. Some of the usernames that you can address messages to are:

NICKOLS	— Kevin Nickols, MS-DOS SIG Manager
DANNYDOS	— Danny Humphress, PCM Managing Editor
LFALK	— Lonnie Falk, PCM's Editor and Publisher
JIMREED	— Jim Reed, PCM's Editorial Director
BOBBYBALLARD	— Bobby Ballard, PCM's DeskMate Expert

All of us will be more than happy to help in any way we can. Keep an eye out for us with the `WHO` command (which shows who is on the system at that time) and say hello.

Delphi employs the Xmodem protocol for program downloading, a very popular eight-bit, error-checking transfer protocol. Many of the terminal packages on the market today support Xmodem. Unfortunately, *DeskMate*, which many of you may be using as a terminal program, does not. But that's no problem. There are several "Free-ware" terminal programs available in the MS-DOS SIG's database that can be downloaded and used in the future when you wish to download other programs.

Another thing I'd like to mention is that you have a choice between two editor utilities on Delphi. The default editor, automatically assigned to new users, is the powerful DEC EDT editor. But EDT is also a complex editor to use. If you're familiar with FILGE (file generator), CompuServe's editor, we are able to set a FILGE-like editor for you to use on Delphi. All it takes is a note from you in the Forum or in Mail indicating your preference. OLDIE, as this editor is called, is easy to learn and use.

On the following two pages we have reproduced a modified version of the Delphi Command Card, with their permission. This card may be removed from the magazine, folded, and kept near your computer as a handy reference to the commands available in the group.

That's "Delphi Bureau" for February. Next month, we'll take a look at how to upload and download files from the MS-DOS SIG. If you, for some reason, have been unable to join us, see Page 34 for complete instructions. The lifetime membership and the first hour of service is entirely free, so sign on and check out the service. I think you'll be glad you did. □

This abbreviated, modified version of Delphi's command card has been created to help our readers who use Tandy® MS-DOS Computers get started quickly on PCM's new MSDOS SIG. It is being reproduced here for your convenience and can be removed, if you wish, and kept near your computer for easy reference.

WELCOME TO DELPHI

Most Delphi commands are self-explanatory. This card will serve as a handy backup reference.

Signing onto Delphi Directly

1. Dial (617)-576-0862.
2. When you have carrier, press [ENTER] once or twice.
3. At "USERNAME" type your membership and [ENTER].
4. At "PASSWORD" type your password and [ENTER].

How To Sign On Using Uninet

1. Dial your local Uninet number.
2. Hit [ENTER][.] [ENTER] at the \ or ".?" prompt.
3. Type DELPHI or GVC at the SERVICE prompt.
4. Then type your USERNAME and PASSWORD as outlined above.

How To Sign On Using Tymnet

1. Dial your local Tymnet number.
2. When "PLEASE TYPE YOUR TERMINAL IDENTIFIER" appears, type A.
3. When "PLEASE LOG IN" appears, type DELPHI.
4. Then type your USERNAME and PASSWORD as outlined above.

How To Sign On Using Datapac (Canada)

1. Dial your local Datapac number.
2. Type [.] for 300 baud or [.] for 1200 baud.
3. Type Set 2:1, 3:126 for full duplex allowing deletes.
4. Type p 1 3106, DELPHI; [ENTER] (Tymnet)
5. Then type your USERNAME and PASSWORD as outlined above.

To obtain your local access number you may call Tymnet at 800-336-0149 or Uninet at 800-821-5340. If you have problems at any time, call Delphi toll-free at 1-800-544-4005. (Mass. 617-491-3393)

Note: Most commands require only enough letters to be entered to make them unique. For example, to enter CONFERENCE from the Main Menu, simply type "C" and [ENTER]. Do not press [ENTER] after commands using the Control Keys. Most other commands require pressing [ENTER] to activate them.

Typing **BYE** from any prompt (except the **MAIL** prompt) will log you off of Delphi.

Typing [?] [ENTER] will generally display a full menu or provide help.

IMMEDIATE COMMANDS (Can be used at any time.)

/HELP — lists Immediate Commands.
/ECHO — turn on character echo.
/NOECHO — turn off character echo. (Used after setting terminal or Uninet or Tymnet node to produce echo.)
/EXIT — exit to next higher menu or command level.
/GAG — turn off incoming /PAGE or /SEND messages.
/LENGTH — shows current number of lines per page on your screen or sets new length.

/NOGAG — turn on incoming /PAGE or /SEND messages after using /GAG.
/PROMPT (1, 2, or 3) — 1=no menu, no explanation; 2=no menu, some explanation; 3=menu plus explanation.
/TIME — show current Eastern time and date.
/WHOIS (username) — shows profile of member (if available).
/WIDTH — shows current screen width format or sets new width.

DATABASE

DIRECTORY — display a directory of all files in the topic.
EXIT — exit database.
HELP — get help on database actions and commands.
READ — read a description of a file. (You must read the file before you download it.)

SEARCH — search a topic by keyword.

SET TOPIC — switch from one topic to another without leaving the database section.

SUBMIT — submit a file for inclusion in a topic. The file must be in your workspace.

WORKSPACE — enter your workspace area.

In order to access a file, you must first **READ** (filename). Once you have read a file, the following actions are available:

DESCRIPTION — displays the file's description again.
DISPLAY — display/list the file on your screen.
DOWNLOAD — use with the buffer capture method of downloading.
EXIT — return to the database prompt.
HELP — get help on commands and actions.
LIST — like display; list a file in an unformatted format.
NEXT — advance to the next group or file. ([ENTER] defaults to NEXT.)
XMODEM DOWNLOAD — download the file using the Xmodem protocol.

WORKSPACE

Workspace is an area for you to store files and messages of all types. This is where you must first upload a file before submitting it to a database. You can file forum messages for retrieval later. Mail messages can be stored here.

From the **MSDOS SIG** prompt type **DA**, and pick a topic, then type **WO** to reach Workspace.

APPEND — append one file to another.
CATALOG — shows which files you have created.
COMMON — go to the Delphi Common work area.
CREATE — creates files and stores it in your area.
DELETE — deletes files you no longer need.
DOWNLOAD — download a file from Delphi to your disk.
EDIT — create and edit your own text files.
EXIT — return to Main Menu.
HELP — explanation of **WORKSPACE** commands.
HOME — return home to your private work area.
LIST — lists contents of any file in your catalog.
PUBLISH — submits your file for Delphi publication in Authors, Members Choice or Newsletters.

PURGE — delete all but current version of duplicate files.
UPLOAD — upload a file from your computer to Delphi.
XDOWNLOAD — download via XMODEM protocol.
XUPLOAD — upload via XMODEM protocol.

MSDOS SIG FORUM

ADD — start a new message thread with a different topic.
BACK — moves backwards within a thread.
DELETE — delete a message.
DIRECTORY — display a directory of messages.
EDIT — edit the current message.
EXIT — exit forum.
FILE — put a copy of a message in your workspace.
FOLLOW — follow a message thread. Read only the messages of a particular thread.
FORWARD — send a copy of a message by mail.
HELP — get help on forum actions and commands.
HIGH — set/show the high message number.
MAIL — take you directly to mail.
NEXT — read next message. ([ENTER] defaults to NEXT.)
READ — read a message. (Typing message number will read that message.)
REPLY — reply to a message.
TOPICS — set/show message topic.

CONFERENCE

EXIT — return to Main Menu.
JOIN (groupname) — join existing group or start new one.
NAME (newname) — change your name or "handle".
PAGE (username) — pages another user in the system.
SCHEDULE — transfer you to the Conference Schedule.
WHO — lists all current users and Conference groups.
Conference Immediate Commands (use while in Conf).
/ACCEPT — accept another's page from within current group.
/ANSWER — respond to or decline **PAGE** from another user.
/CANCEL — terminate a **PAGE** to another user.
/EXIT — like **CONTROL-Z**; gets you out of wherever you are.
/GAG — disable /SEND's from people outside your conference group.*

/GLOCK — lock the group's attributes.*
/GNAME (newname) — change current group name.
/GPASS (password) — select a group password.*
/GPRIVATE — make the group private.*
/GQUIET — makes the group have silent entry and exit.*
/HELP — get help on conference actions and commands.*
/JOIN (groupname) — join an existing group.
/LOG — save a transcript of your conference in your workspace.*
/MAIL — takes you directly to Mail.
/NAME — create a conference nickname (handle).
/PAGE — ask another user to join your group.
/PASS (password) — say the password for admittance into password groups.
/REJECT — a pleasant "No thank you" to whomever is paging.
/REPEAT — turns Echo on or off.*
/RNAME (nickname) — show the username of a person using a handle.
/SCHEDULE — transfer you to the Conference Schedule.
/SEND (username) — send message to current user.
/SQUELCH (username) — ignore messages from a user.*
/TALK — like /JOIN, but doesn't leave current group. /# also works, where # is the number of a conference group. Allows you to participate in more than one group at the same time.
/WHO — lists all current users and Conference groups.
/WHOIS (username) — displays (username) Profile.

*Note: Many of these commands may be preceded by NO. For example, /GAG disables sends, but /NOGAG resumes them.

DELPHI MAIL

Primary Mail Menu (DMail)
CATALOG — lists all Mail files you have created.
EXIT — return to Main Menu.
HELP — explanation of Mail commands.
MAIL — send or read mail. Enters Secondary Mail Menu.
SCAN — display the headers for all unread mail.

Secondary Mail Menu (MAIL)
[ENTER] — depressing the return or carriage return key, will read the next message or more of the current message.
BACK — displays previous message.
DELETE — deletes current (last read) message.
DIRECTORY — lists summary of your mail messages.
DIRECTORY / Folder — lists folder names.
DIRECTORY (folder name) — lists summary of messages in the specified folder. For instance, DIR Pending.
EXIT — returns to Main Menu.
EXTRACT (filename) — adds current message to named file.
FILE (folder name) — adds current message to the named folder.
FORWARD — forward present message to others.
NEXT — skips to next Mail message. ([ENTER] defaults to NEXT.)
READ — displays your Mail messages.
READ (folder name) — reads contents of named Mail folder.
READ (n) — allows you to read selected message number.
READ /NEW — for new MAIL arriving while in MAIL.

DELPHI TERMINAL CONFIGURATION GUIDE

8 bit ASCII*
 1 stop bit*
 no parity*
 asynchronous full-duplex
 no auto-linefeed or carriage-return linefeed
 XON-XOFF or Handshaking should be enabled

*sometimes you have to experiment with other combinations such as: (7 bit, 1 stop, noparity) or (8 bit, 1 stop, even or odd parity).

NOTES

To erase a character, Delphi uses the ASCII delete/rubout key which is decimal 127. If necessary, the terminal program should translate the backspace key to a delete/rubout. Unfortunately the networks do not echo the delete/rubout correctly however it will have the desired effect.

Delphi uses the following control characters:

CONTROL-Z — end of input or exit to next higher menu.
CONTROL-S — suspends sending.
CONTROL-Q — resumes sending.
CONTROL-O — skips to end of file or message.
CONTROL-U — cancels input for current line.
CONTROL-R — redisplay current line.
CONTROL-X — cancels everything typed ahead but unsent.
CONTROL-C — cancel current activity and start over.

If a particular control key is causing the terminal program to take some other action, then the terminal program should be reconfigured to use any of the other available control keys in place of the one required by Delphi.

Modified Command Card for PCM Readers

DELPHI™

Delphi is a © trademark of General Videotex Corporation.
 PCM is a © trademark of Falsoft, Inc.
 Tandy is a © trademark of Tandy Corporation.
 Tymnet is a © trademark of Tymshare, Inc.
 Uninet is a © trademark of Uninet, Inc.

REPLY — sends a reply to sender of current message.
SEARCH (string) — searches current Mail file for specified character string.
SELECT — pick messages for delete operation.
SEND — sends message to another user or users.
SEND (filename) — sends file (filename) to other user(s).
SEND/EDIT — calls editor to edit message being sent.
SEND/LAST — uses last message as text for current message.

HELP

Contains a full description of all Delphi services using the same structure as the Delphi Menus.

PEOPLE ON DELPHI

Enter information about yourself; find out about others.

I-AM — add or change information about yourself.

ADD — adds to existing information.

CHANGE — removes all current information about you and request new info.

DELETE — deletes all information under a given keyword heading.

DISPLAY — prints your personal profile.

EXIT — returns to Main Menu.

WHOIS (membername) — displays member profile if available.

SEARCH — find members with particular interests.

BROWSE — browse through member profiles.

LIST-KEYWORDS — shows keywords used in member profiles.

This section is accessed from the Delphi Main Menu. When someone does a /W command in conference, this is the information that will be displayed about a member.

USING-DELPHI

ADVICE FROM DELPHI — answers to most frequently asked questions.

CREDIT POLICY — explanation of current DELPHI policy.

GUIDED-TOUR — a brief version of the tour you took at signon.

MAIL TO SERVICE — send comments and suggestions to DELPHI.

NETWORK-INFO — phone numbers and login procedures for data networks.

PREMIUM-SERVICES — information concerning the extra cost services.

RATES-AND-PRICES — official Delphi rates and prices.

SETUP — terminal and network configuration.

LENGTH — lets you find your screen length and tailor Delphi accordingly.

MENU — choose default menu at sign-on.

PASSWORD — change your password. (frequent changes are recommended.)

PROMPT — select level of menu prompting desired.

SET-TYMNET — experiment with setting network parameters.

TERMINAL — special features for DEC VT100 and VT52 users.

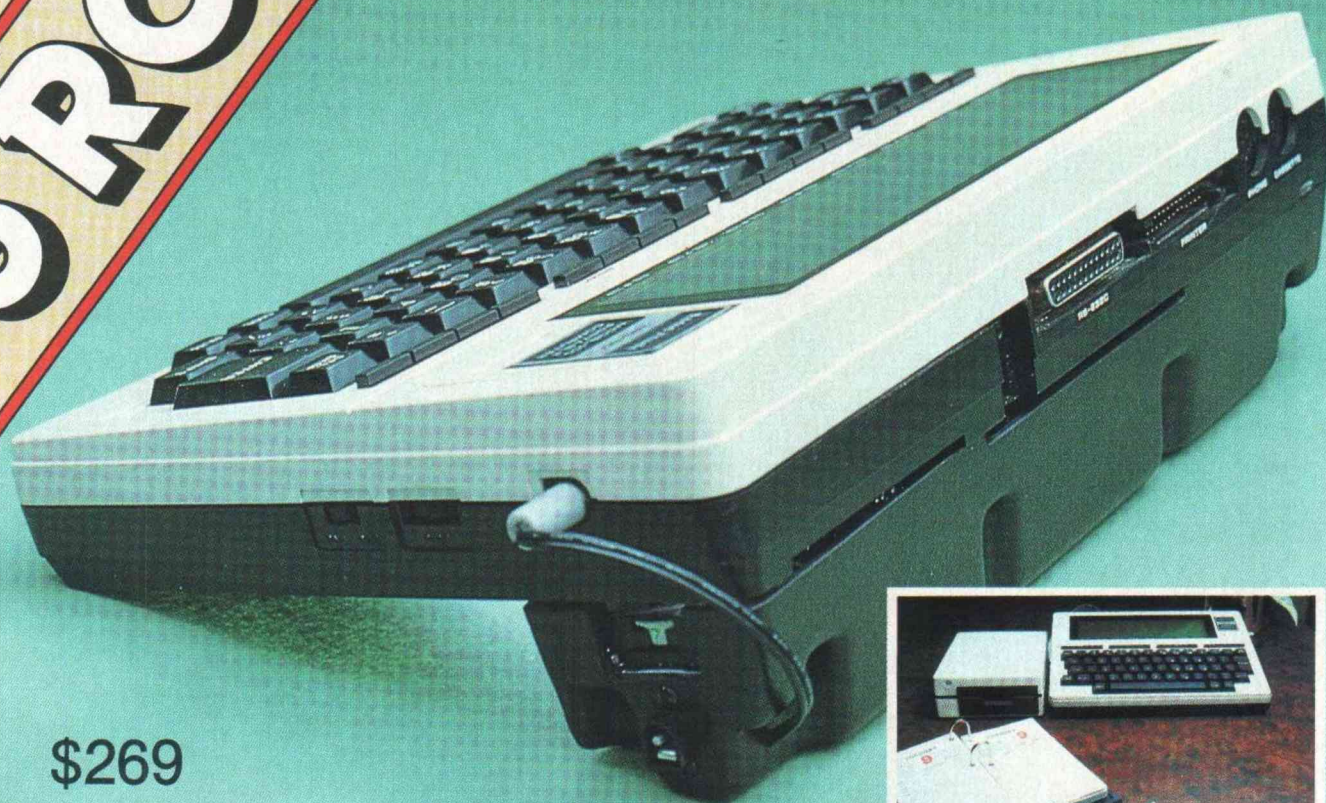
WIDTH — tailor Delphi to fit your screen width.

USAGE-HISTORY — view your to-date activities on Delphi.

6 ROM BANK

plus powerful Nicad battery pack that gives 30 hours of power to your Model 100

6 ROM



\$269

A joint PCSG/CRYPTRONICS project.



Now you can access LUCID, DISK +, WRITE ROM, RAM+ and others instantly.

The 6 ROM BANK plus battery pack lets you have the ultimate Model 100 system. When you get your unit just press in up to 6 ROMS into the sockets then push in a simple plug into your Model 100's ROM socket. The ROM bank has two posts on either end that insert into the two little holes on the underside of your Model 100.

The ROM bank props up the Model 100 at the same angle and height as those little legs you've seen. The ROM BANK itself is only about 1½" deep and it runs the width of your Model 100. It only weighs one pound. It not only installs instantly, but it pops free in a second if you need everything to lie flat in a briefcase.

Change from ROM to ROM with the touch of a thumb switch.

You can go from LUCID to WRITE to DISK+ to any other ROMS just by turning the thumb switch at the side of the ROM bank. The 6 ROM BANK is a sturdy well built construction that looks like it is a part of your Model 100.

What is also fantastic is that the ROM bank has a powerful NICAD battery and recharger built right in. This power source supplies up to 30 hours of life to your Model 100 with just 6 hours of recharge. What's nice is that it recharges right from your Model 100's power adapter. This is a quick charge system and if you need power in a hurry, you can get 6 hours of life for your Model 100 by just charging for an hour and half.

Suddenly, the Model 100 is a very powerful computer.

If you have the HOLMES portable disk drive with its powerful bundled database software package or the CRYPTRONICS 128K RAM expansion along with LUCID, WRITE ROM and DISK+ in the 6 ROM bank with its rapid rechargeable NICAD power source, you have the ultimate portable system. It's all available only from PCSG.

As usual we don't want you to take our word for it. The 6 ROM BANK is sold on a 30 day trial. If you are not satisfied simply return within thirty days for a full refund. MC, VISA, AM. EX. or C.O.D.

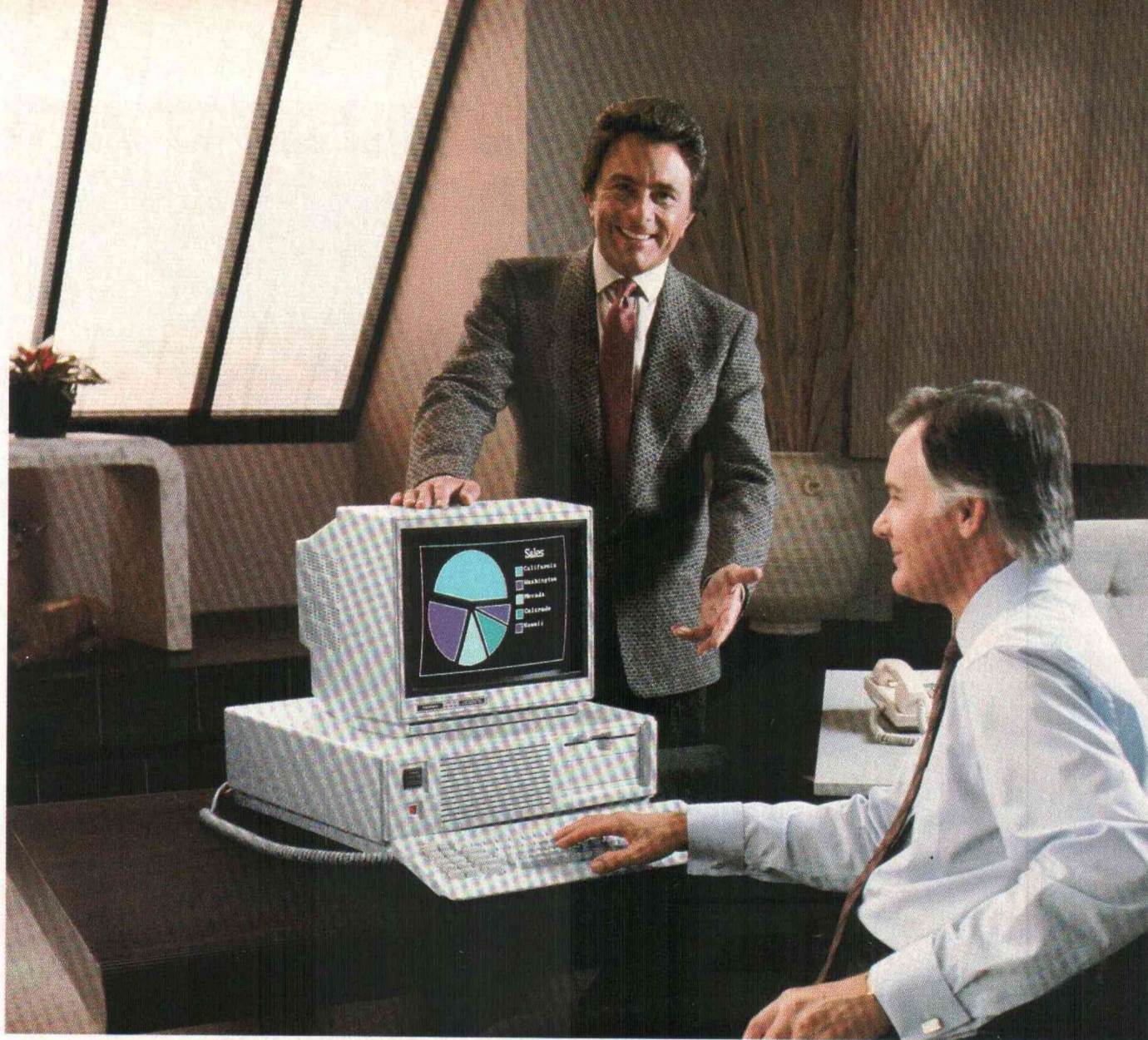
1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

VISA / MASTER CARD / AM. EXP. / C.O.D.





The New Tandy® 3000

The difference is power
...and affordability.

Introducing the Tandy 3000, the affordable alternative to the IBM® PC/AT. Here's the power you need to manage your business, to network computers, or to create a multiuser system.

Unmatched compatibility

The Tandy 3000 uses the advanced MS-DOS 3.1 operating system. And since the Tandy 3000 is compatible with programs designed for the PC/AT, as well as the PC/XT, it cuts through today's software confusion. Choose from advanced word-processing packages, database management and powerful accounting software.

Power to share

The Tandy 3000 is also designed to use the forthcoming XENIX 5.0 multiuser operating system. Two to six people in an office can use the 3000 simultaneously with low-cost data display terminals.

In offices already equipped with MS-DOS computers, the Tandy 3000 is the link that brings them all together. Using our ViaNet local area network, the Tandy 3000's high speed is available to all network users. The Tandy 3000 is your affordable solution for office automation.

Designed for High Performance

The Tandy 3000 comes with an 80286 microprocessor for the ultimate in speed and performance. Its 16-bit architecture operates at 8 megahertz—twice that of the industry standard.

The Tandy 3000 comes with one-half megabyte of main memory (512K RAM). This memory can be expanded to 640K without using an expansion slot. Total internal memory can exceed an astounding 12 megabytes (under XENIX).

As for storage, the Tandy 3000 (25-4001, \$2599) is equipped with a high-capacity, 5 1/4" slim-line floppy disk drive. For the utmost in compatibility, this drive can read 1.2-meg and 360K formats for use with any IBM PC diskettes.

For maximum storage capacity, choose the Tandy 3000 HD (25-4010, \$3599) with a built-in 20-megabyte hard disk drive for fast access to volumes of data.

Both versions of the Tandy 3000 feature a built-in real-time clock for time-sensitive applications and a serial/parallel interface.

The Tandy 3000 has ten expansion slots, including seven IBM PC/AT-compatible slots, two PC/XT-compatible slots and a PC/XT-compatible half-slot for the serial/parallel interface board.

Tandy...Clearly Superior™

High performance, compatibility, multiuser and networking capabilities, and remarkable expandability: the Tandy 3000 has the power to put you in command.

Available at over 1200
Radio Shack Computer Centers and at
participating Radio Shack stores and dealers.

Radio Shack®
COMPUTER CENTERS

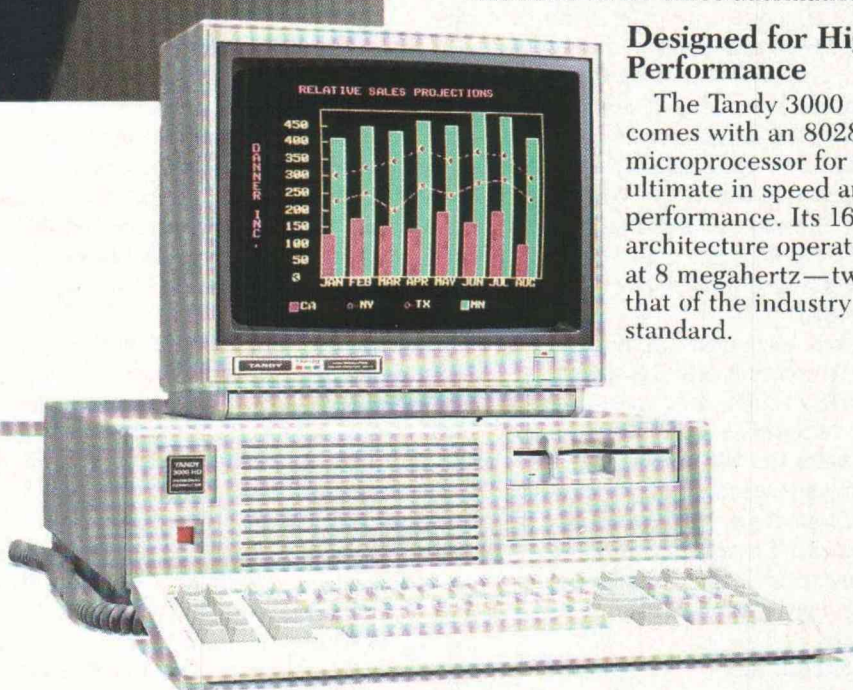
A DIVISION OF TANDY CORPORATION

Send me a Tandy 3000 brochure.

Dept. 86-A-992, 300 One Tandy Center
Fort Worth, Texas 76102

Name _____
Company _____
Address _____
City _____
State _____ Zip _____
Phone _____

Prices apply at Radio Shack Computer Centers and at participating stores and dealers. MS-DOS and monitor sold separately.
IBM/Registered TM International Business Machines Corp. MS and XENIX/TM Microsoft Corp. ViaNet/TM VIANETIX.



*A tour of the MS-DOS
compatibility bridge, ANSI.SYS*

What is ANSI.SYS?

By John B. Harrell, III

Have you noticed that some software uses ANSI.SYS and insists that you must install a special command in your configuration file? Ever wondered why they use this program? I am going to delve into this mysterious and powerful utility which, unfortunately, lacks a great deal of documentation.

Last month, I showed you how to use direct memory accessing to really "jazz" up your programs. This works great if you will always run it on similar machines. If you want to make it compatible with most IBM PC compatibles, you can use the ROM BIOS video interrupt calls to handle all input and output for your program. The degradation in performance is noticeable.

If you decide to make your masterpiece compatible with all MS-DOS computers, you must resort to performing all of your video and keyboard accesses via standard operating system functions. Unfortunately, these MS-DOS functions do not provide adequate services such as positioning the cursor, clearing the screen and others. Here's

where ANSI.SYS enters the picture.

ANSI.SYS is a program file that is loaded when the system is initialized. The command in your CONFIG.SYS file (DEVICE=ANSI.SYS) tells the system to load this utility program and execute the initialization code. This code links the ANSI.SYS code into the system's video and keyboard drivers and then terminates, leaving ANSI.SYS installed as a part of the operating system.

The ANSI.SYS driver contains many special features that enable you to perform just the functions you need. This driver allows you to change graphics modes, set and reset colors, reposition the cursor and *redefine any key on your keyboard!*

You cannot use these function code sequences from the BASIC interpreter or compiled BASIC code. You *must use* the specified MS-DOS functions, and BASIC bypasses the MS-DOS system to perform direct screen and keyboard handling. Even if you program only in BASIC, *read on!* There are several other ways to use this mysterious program and I am going to explain them.

Extended Screen and Keyboard Control

Figure 1 contains a list of the extended screen and keyboard control sequences specified for the IBM-PC and most compatibles. These codes even work with the really incompatible com-

puters such as the Zenith 100 and the Wang PC. Computers that do not implement a BIOS similar to the IBM-PC usually provide their own BIOS functions that use a superset of the ANSI control codes. Look in your computer's DOS reference manual if you are not sure.

While you study Figure 1, let me stress one item. All of the codes used in these ANSI control sequences are ASCII characters. Specifically, this means that numbers *must be* passed as an ASCII character string; for instance, the number 12 must be contained in the control sequence as the character '1' followed by the character '2'. This is an important concept.

Looking at Figure 1, you will notice there are several different cursor positioning commands. The "Cursor Position" command moves the cursor to the specified row and column. The default value for omitting a parameter is '1', and omitting both parameters will "home" the cursor to the upper-left corner of the video. The "Horizontal and Vertical Position" command duplicates this command.

ANSI.SYS also provides commands (Cursor Up, Down, Forward and Back) that allow you to move within the column or row without having to specify extra information. As in the previous command's discussion, the default value of an omitted parameter is '1'.

(John B. Harrell, III has written for microcomputer magazines for three years. He holds a bachelor's degree in computer science and is a software technical expert for Navy electronic support measures systems.)

The next two commands, "Cursor Position Report" (CPR) and "Device Status Report" (DSR) work together. Issuing a DSR sequence will cause ANSI.SYS to respond with a CPR on the MS-DOS standard input device. This is the sequence that must be used to read the current location of the cursor. For example, if the cursor is currently located in Column 30 of Row 12, issuing a DSR sequence will return

the following sequence of characters to the input device: ESC[12;30R.

The "Save Cursor Position" and "Restore Cursor Position" commands can be used to real advantage. The save command sequence causes ANSI.SYS to save the location of the current cursor internally. You may then reposition the cursor and perform any operation on the video. You have only to issue the restore command sequence to return the

cursor to its exact starting location. I'll present a detailed example of this later.

The "Erase in Display" command is the equivalent of the BASIC CLS statement and will clear the screen and leave the cursor in the home position. The "Erase in Line" command will erase from the current cursor position to the end of the line.

The "Set Graphics Rendition" (SGR), "Set Mode" (SM), and "Reset

Figure 1

ANSI.SYS Control Sequences

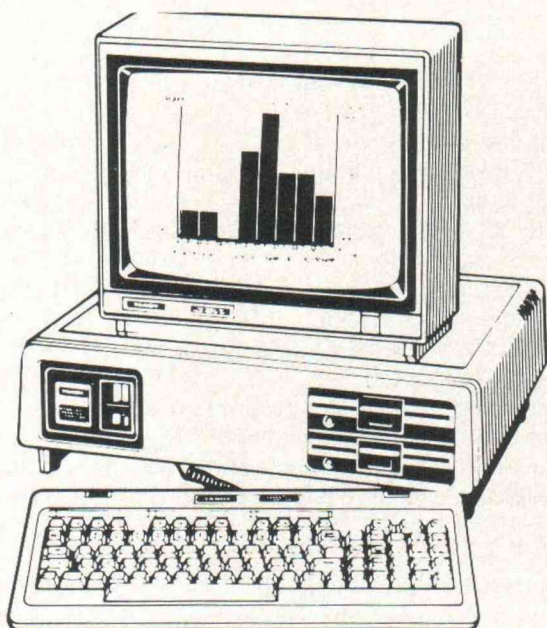
Command Codes	Function of Command
ESC[<i>#</i> ; <i>#</i> H	Move the cursor to the position indicated by the row and column (<i>#</i> ; <i>#</i>) parameters. Both parameters default to '1' if omitted.
ESC[<i>#</i> ; <i>#</i> f	Horizontal and vertical position — same as above.
ESC[<i>#</i> A	Move the cursor up ' <i>#</i> ' rows (default '1') without changing the column location. It is ignored if the cursor is at the top of the screen.
ESC[<i>#</i> B	Move the cursor down ' <i>#</i> ' rows (default '1') without changing the column location. It is ignored if the cursor is at the bottom of the screen.
ESC[<i>#</i> C	Move the cursor forward ' <i>#</i> ' columns (default '1') without changing the row location. It is ignored if the cursor is at the right edge of the screen.
ESC[<i>#</i> D	Move the cursor back ' <i>#</i> ' columns (default '1') without changing the row location. It is ignored if the cursor is at the left edge of the screen.
ESC[<i>#</i> ; <i>#</i> R	Specifies the format for the Cursor Position Report issued in response to a Device Status Sequence (see below). The information is reported via the standard input device. The first parameter specifies the current line and the second parameter is the column.
ESC[6n	Issues a Device Status Report to the console driver and requests a Cursor Position Report in reply (see above).
ESC[s	Internally saves the cursor position.
ESC[u	Restores the cursor position saved with the previous command.
ESC[2J	Erase in Display command — clears the screen and homes the cursor to the upper left corner.
ESC[2K	Erase in Line command — erases the current line from the cursor position.
ESC[<i>#</i> ; <i>...</i> ; <i>#</i> m	Set Graphics Rendition (SGR) — sets the character attribute specified by the parameters as listed below. These attributes will remain in effect until the next SGR command is issued:

Parameter	Meaning
0	All attributes off (black on white)
1	Bold on (high intensity)
4	Underscore on (monochrome only)
5	Blink on
7	Reverse video on
8	Canceled on (invisible characters)
30	Black foreground
31	Red foreground
32	Green foreground
33	Yellow foreground
34	Blue foreground
35	Magenta foreground
36	Cyan foreground
37	White foreground
40	Black background
41	Red background
42	Green background
43	Yellow background
44	Blue background
45	Magenta background
46	Cyan background
47	White background

NOCONA

ELECTRONICS

TRS-80 Model 2000 by Tandy®



TRS-80® Model 100 by Radio Shack



MS-DOS Based

Computer System

256K MOD2000 W/MONO MONITOR	1019.00
256K MOD2000 W/COLOR MONITOR	1299.00
256K MOD 1200 HD 2 DR.	1499.00
128K MOD 1000 W/MONO 1 DR.	779.00
128K MOD 1000 W/COLOR 2 DR.	1119.00
256K MOD 1000 W/HARD DISK	1439.00
24K MOD 100 PORT	399.00
24K MOD 200 PORT	679.00
64K MOD 4D 2 DR	839.00
512K MOD 6000 2DR.	3099.00
512K MOD 6000 w/15MEG HD	3779.00
15 MEG HARD DISK	1119.00
35 MEG HARD DISK	2059.00
DMP 105 PRINTER	145.00
DMP 2200 PRINTER	1149.00
DMP 130 PRINTER	249.00
DMP 430 PRINTER	585.00
DMP 2100P PRINTER	1019.00
CGP 220 COLOR PRINTER	449.00
DWP 510 PRINTER	999.00
TRACTOR DWP 210	99.00
TRACTOR DWP II/510	159.00
TRACTOR FEED DMP2100P	99.00
PARALLEL PRINTER SWITCH	96.00
PARALLEL PRINTER CONTROLLER	179.00
PARALLEL PRINTER CABLE	29.00
DISK VIDEO INTERFACE	399.00
32K MOD 600 PORTABLE	1149.00
512K MOD 3000	1819.00

100% RS COMPONENTS NO FOREIGN DRIVES OR MEMORY FULL WARRANTY

ALL RS SOFTWARE 20% OFF CATALOG PRICE

CASHIERS CHECK OR MONEY ORDER MUST ACCOMPANY ALL ORDERS

(817) 825-4027

NOCONA ELECTRONICS • BOX 593 • NOCONA, TX 76255

Figure 1
Continued

ESC[=#h

Set Mode (SM) — Invokes the screen width or type specified by the parameter:

Parameter	Meaning
0	40 x 25 black and white
1	40 x 25 color
2	80 x 25 black and white
3	80 x 25 color
4	320 x 200 color graphics
5	320 x 200 black and white graphics
6	640 x 200 monochrome graphics
7	Wrap at the end of the screen line
-	— Special Tandy 2000 Modes —
8	640 x 400 color graphics
9	640 x 400 black and white graphics
10	1640 x 200 color graphics

ESC[#1

Reset Mode (RM) — same as the SM command sequence with the exception that Mode 7 disables the word wrap at the end of the current screen line.

ESC[#;#;...;#p or
ESC["string"p or
any combination
of strings and
decimal numbers

Keyboard Key Reassignment — The first code or string parameter value defines the key to be reassigned unless this code is NUL (zero). Then the first and second codes make up the extended ASCII definition of the key to be redefined. The remaining characters or codes in the command string define the sequence of ASCII codes that will be generated whenever this key is pressed.

Mode" (RM) control sequences are used to set and control the various video modes and colors that can be displayed. Let's begin with the last two sequences — the set and reset mode commands.

When you look in Figure 1 for the character sequences for these commands, you will see four different command character sequences. They are all equivalent — just use the one that makes you feel most comfortable. Figure 1 also contains a brief description of the modes set by the parameters. Tandy 2000 additional screen modes are listed below the line (by the way, there is a typographical error in the character strings for the SM command in the *MS-DOS Programmer's Reference Manual*, Catalog No. 26-5403 — correct it as listed in Figure 1).

The only difference between the SM and RM commands is in the handling of the "wrap-around" mode at the end of the line. SM sets wrap-around mode when Mode '7' is used, and RM resets this mode, throwing away characters past the screen's end-of-line.

The last control sequence is for keyboard key reassignment. The first code (or first two codes if the first code is NUL) specify the key to be reassigned. The remainder of the command may be composed of code numbers representing characters or a character string in

quotes or any combination of these codes. The code numbers are the decimal number representation of the ASCII character code generated when the key is pressed. A table of key values may be found in your DOS reference manual or the *Programmer's Reference Manual*.

Program Listing 1 is a short program written in *Turbo PASCAL* which assigns useful commands to the F7 through F10 function keys. I found this the quickest approach for changing your keys but there are several other methods for using ANSI.SYS and I will describe each of them.

If you use *Turbo*, you must use the 'P' compiler directive to force *Turbo* to use MS-DOS's standard output file (use normal MS-DOS function calls for output instead of the BIOS code). This method lets you quickly and easily build a program that will change your whole keyboard layout, and you can include it in your AUTOEXEC file so you never have to remember it. You really should not reassign the F1 through F6 keys as these are used by the MS-DOS command processor for command line editing.

Doing it Without a Program

There are three methods for issuing ANSI.SYS commands without using a

program. If you have written for and received my MENU demonstration disk, you have already experienced these two methods — using the TYPE and PROMPT commands. The third method uses the ECHO command.

The READ-ME file on the demonstration disk contains a banner which is written in reverse video using ANSI.SYS control codes. You may enter these codes using any text processor you desire as long as it allows entry of control codes. The quickest available tool is EDLIN. Just as a reminder, if you use EDLIN, you enter control codes by typing a CTRL-V followed by the character.

After you have completed building the file with the codes that you desire, write it to the disk. To display the text, issue the following command:

```
TYPE filename.ext
```

where "filename.ext" is the name that you assigned to the file. This will cause the text to be rapidly displayed on the video screen.

You can also use the DOS ECHO command from a batch file. In order for this to work properly, you must first issue an ECHO OFF command. See Figure 2 for a sample batch file.

In Figure 2, the batch file uses the

ECHO command to clear the screen and set a title at the top of the video screen using inverse video. Line 1 of the batch file turns command echo mode off so that the DOS commands are not displayed on the video as they are executed.

Line 2 clears the video screen. Lines 3 through 5 position the cursor to Column 60 of the first three lines of the video screen and write a block there containing the program title. Line 6 positions the cursor to Column 1 of Line 20 and ends the batch file. The characters “^[]” represent the ASCII ESC (decimal 27) character.

The third method of using these control code sequences involves using the DOS PROMPT command. Figure 3 contains an excerpt from the DOS reference manual listing the appropriate PROMPT special strings that may be used. In combination with the ANSI.SYS control sequences, you can really enhance your displays.

Again, if you have received a copy of the MENU demonstration disk, you have already seen one of these examples. My AUTOEXEC file initially sets the system prompt to display the time and full DOS path name in high intensity video prior to returning the command line to normal video mode. I accomplished this using ANSI control sequences and special prompt strings in the following command:

```
PROMPT $e[1m{$t$h$h$h} $p$g$e[0m
```

Using Figures 1 and 3, let's decipher this command. First of all, \$e generates an ESC character so \$e[1m tells DOS to set high intensity mode, and \$e[0m returns the command line to normal intensity. The \$t displays the current system time and the \$h\$h\$h\$h backspaces over the hundredths of seconds. The \$p displays the path name followed by the normal DOS > symbol generated with

the \$g string. The braces ({}) and spaces are displayed as “natural” characters where they appear in the prompt string.

Now let's examine a more complex example that saves the current location of the cursor and positions the cursor to the top line of the screen. There, the current system date and time are displayed in high-intensity reverse video mode. The cursor is then positioned under the date and time and the current directory path is displayed. Finally, the video mode is returned to normal, the cursor returned to its initial position and the normal DOS prompt is displayed.

This prompt is long and is printed here on more than one line. Make sure that you enter it at the DOS prompt as one continuous string of characters. Note that the case (upper- and lower-case) of the characters in this prompt string is very important:

```
PROMPT $e[$e[1;7m$e[1;40H$e[K
$d $t $e[2;40H $e[K Current
Directory: $p $e[0m$e[u$n$g
```

Disect this string piece-by-piece just as we did before and you will see just how useful this technique can be. One word of caution — you cannot use the PROMPT command to achieve immediate effects in a batch file. You must use the ECHO command or display a file contents on the video if you want to change video modes in the middle of a batch file.

This limitation arises because of the way PROMPT works. It only is effective when the DOS command prompt would normally be displayed waiting for user input. This normally means at the end of the batch file.

Conclusion

These techniques may be used to extract the most from your computer system. You may write programs for any computer using the MS-DOS operating system using these methods and be guaranteed that your masterpiece will work properly.

Like anything else, there is a trade-off. Using my direct access methods (from the last column) provides an exceptionally rapid method of displaying information on the video screen, but you are limited in compatibility. Using the ANSI.SYS driver means that you can run on almost any machine that uses MS-DOS, but you pay a heavy penalty in display speed. The application must ultimately determine the proper method to be used. **PCM**

Figure 2

Sample Batch File Using ANSISYS Codes

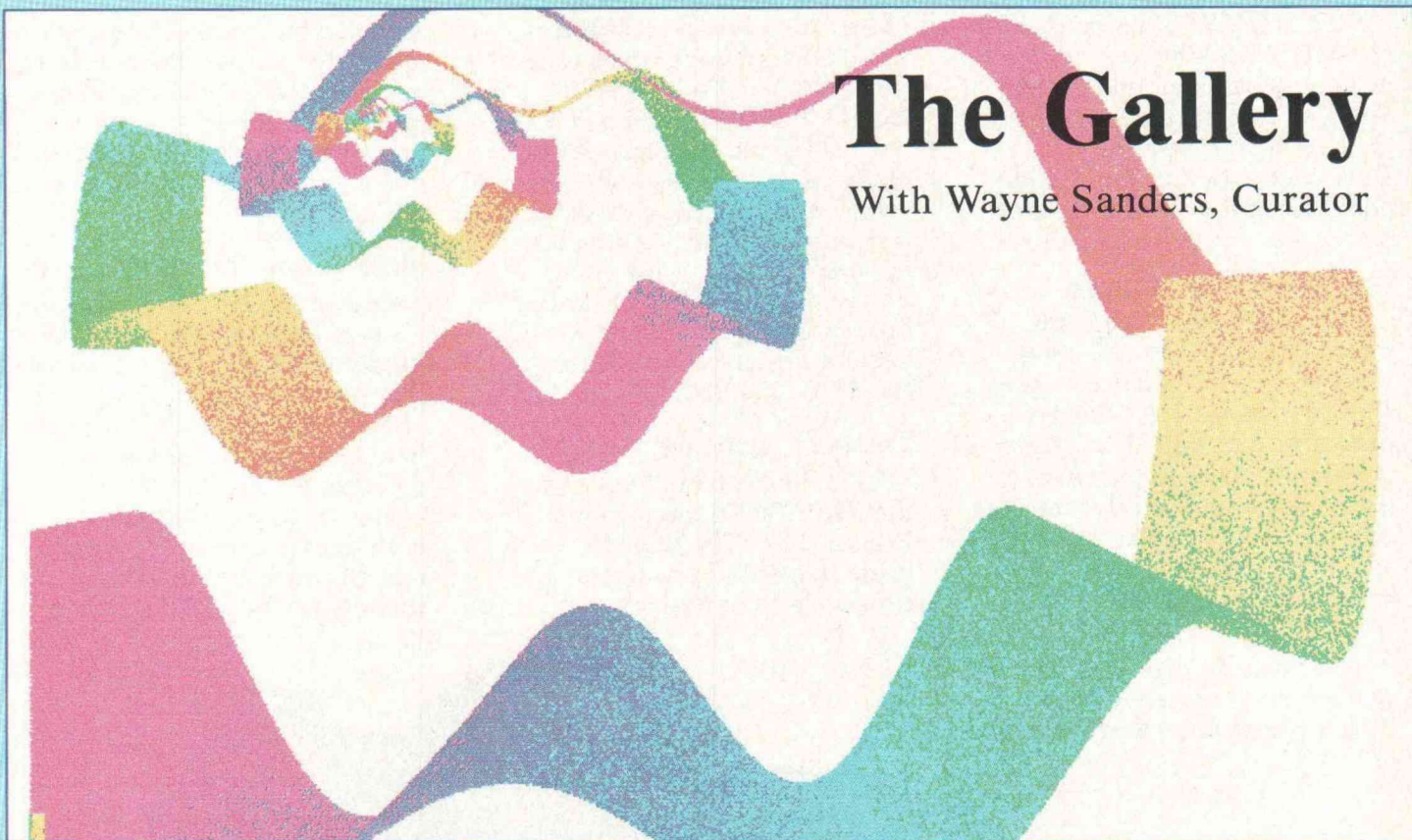
```
echo off
echo ^[[2J
echo ^[[1;60H^[[7m          ^[[0m
echo ^[[2;60H^[[7;1m    Program Title  ^[[0m
echo ^[[3;60H^[[7m          ^[[0m
echo ^[[20;1H
```

Figure 3

Special Codes for PROMPT Command

These special prompt characters are used with the DOS PROMPT command to display the information listed below. A special prompt character is constructed by preceding one of the characters from the table below with a '\$':

Character	Prompt Meaning
\$	Displays the '\$' character as part of the prompt string
t	Displays the current system time
d	Displays the current system date
p	Displays the current directory path name
v	Displays the MS-DOS version number
n	Displays the default disk drive specification
g	Displays the '>' (greater-than) symbol
l	Displays the '<' (less-than) symbol
b	Displays the ' ' (vertical bar) symbol
-	Displays a carriage return and line feed sequence
s	Displays a leading space
h	Backspaces the cursor one location
e	Displays the ESC character (ASCII decimal value 27)



The gallery exhibit this month demonstrates using mathematical formulas to create exciting displays, a color-mixing technique called "dithering" and "palette animation." The program came to us from Colin Foster of Las Vegas, Nevada.

This program will run as-is on a Tandy 1000. Change the first line of the program to read MODEL=2000 if you are using a Tandy 2000. It takes several hours to run, so be patient — the results are stunning.

The path of the "ribbon" can be changed by altering the formula in Line 1120. Some suggested formulas are listed in Figure 1.

If you would like to have your graphics creation presented here, send it in. A winning gallery exhibit is chosen each month and the "artist" is awarded \$50. Address your entries to PCM Gallery, P.O. Box 385, Prospect, KY 40059.

Figure 1
Replacement formulas for Line 1120

```
X=Q*COS(A)*COS(A)+X0:Y=Q*SIN(A)/2+Y0
X=2*COS(A)+X0:Y=Q*SIN(A/3)/2+Y0
X=Q*TAN(A)+X0:Y=Q*SIN(A)/2+Y0
```

The listing:

```
1000 MODEL=1000
1010 IF MODEL=2000 THEN SCREEN 3,1:XM=640:YM=400
1020 IF MODEL=1000 THEN CLEAR,,,32768:SCREEN 5,1:XM=320:YM=200
1030 KEY ON:KEY OFF
1040 DEFINT C,D,P,Q:DIM P(23)
1050 XM=320:YM=200:PI=3.14
1060 X0=XM/4:Y0=YM/8:J=2*PI/11:B=J:R=1
1070 FOR Q=1 TO 16:READ P(Q):NEXT
1080 DATA 0,13,12,14,10,11,9,5,13,12,14,10,11,9,5,0
1090 PALETTE USING P(1)
1100 WHILE R<550
1110   FOR Q=W TO R
1120     X=Q*COS(A)+X0:Y=Q*SIN(A)/2+Y0
1130     IF X<1 OR X>XM OR Y<1 OR Y>YM THEN 1180
1140     IF K AND POINT(X,Y) THEN 1170
1150     DC=INT(RND*(A-A0)/J)+C : IF DC=8 THEN DC=1
1160     PSET (X,Y),DC
1170   NEXT Q
1180   A=A+2*PI/(R*10):W=R*(1-ABS(.4*SIN(PI/4+A/2)))
1190   K=INT(A/PI)/2>INT(INT(A/PI)/2)
1200   IF A>B THEN B=B+J:A0=A:C=C+1:IF C=8 THEN C=1
1210   R=R+.08:X0=X0+.03:Y0=Y0+.03+SIN(7*A)/2
1220 WEND
1230 WHILE INKEY$=""
1240   FOR Q=1 TO 7
1250     SWAP P(Q),P(16)
1260     PALETTE USING P(Q)
1270     SWAP P(Q),P(16)
1280   FOR Z=1 TO 50 : NEXT Z
1290 NEXT Q
1300 WEND
```

PCM

Enhance your Tandy Portable

Your Radio Shack Model 100 (26-3802, \$499.00) or Tandy 200 (26-3860, \$799.00) are the heart of a complete computer system. We back both machines with all the extensive accessories to make them not only convenient portables, but powerful desktop tools, too.

Get rapid disk storage

The Tandy 200/Model 100 Portable Disk Drive (26-3808, \$199.95) gives you fast access to 100K of data on 3 $\frac{1}{2}$ " floppy diskettes. And since it's battery powered*, you don't have to leave your important documents, reports, spreadsheets, statistics and other vital data back home. You can take them with you wherever they're needed.

This portable drive may be small, but it's not a toy. Menu-driven operation makes it easy to

use. You can get a complete list of files for any disk. It features these powerful operating system functions: FORMAT, SAVE, LOAD, RENAME and BACKUP. A 19,200-baud data-transfer rate easily beats the speed of cassettes. And you can choose battery operation, or just attach an optional adapter to use AC power. And for today's traveling businessperson, there's a convenient Portable Drive Case (26-3815, \$24.95).

Portable printing

Like the Portable Disk Drive, the TRP 100 Thermal Ribbon Printer (26-1275, \$299.95) is made to be used anywhere—it runs on both batteries* or AC

power, and it only weighs 5 $\frac{1}{2}$ pounds. Heat from the super-quiet thermal head transfers high-contrast characters onto plain 8 $\frac{1}{2}$ " paper. You can also use thermal roll paper without the ribbon for direct transfer.

High-speed data communications

One of the most exciting features of both the Tandy 200 and Model 100 is the built-in modem that lets you communicate with other computers over the phone line. Our Direct-Connect Modem Cable (26-1410, \$19.95) comes with one free hour on both the CompuServe Information Network and the Dow Jones News/Retrieval Service.



*Batteries extra. Prices apply at Radio Shack Computer Centers and participating stores and dealers.

with our line of accessories...

Access news, weather, sports, business reports and more. Plus, you can communicate with your home or office desktop computer. It's a whole new world of applications power!

When you're on the road, you can bypass direct connection with our Acoustic Coupler (26-3805, \$39.95). It's a great way to keep in touch with your home office. And with our padded carrying case (26-3811, \$39.95), we make portability all it's supposed to be.

Versatile Bar Code Reader

The useful Bar Code Reader (26-1183, \$99.95) is an optical-scanning device that's ideal for many uses, including inventory control, billing and couponing. Software drivers, callable from BASIC, allow you to read Universal Product code, 3 of 9, Plessey and other bar-code formats.

Real desktop power

If you need more memory, your Model 100 can take up to three 8K RAM modules (26-3816, \$59.95 each) for a total of 32K. And the Tandy 200 can take up to two 24K RAM modules (26-3866, \$199.95 each) for a total of 72K.

It's easy to convert your Model 100 laptop computer into the perfect desktop computer. Just pop in Model 100 legs (26-3812, \$3.99) for the best viewing angle. Then plug in your computer using an optional U.L. listed AC adapter (26-3804, \$5.95).

Tandy . . . Clearly Superior™

Stop by your local Radio Shack Computer Center and see the advanced technology 24K Model 100 and 24K Tandy 200, and take a look at our complete line of accessories and over 25

software applications. With our commitment to advanced technology, you'll find everything you need for professional, portable computing power—for your desk or on the road.

Radio Shack®
The Technology Store™

A DIVISION OF TANDY CORPORATION

Send me a 1986 Computer Catalog.

Mail To: Radio Shack,
Dept 86-A-136
300 One Tandy Center
Fort Worth, Texas 76102

Name
Address
City
State Zip
Phone



*Build an emergency
power supply for your
Tandy 600*

Emergency Power for your Tandy 600

By Carl Oppedahl

There you are with your laptop computer in a Lockheed L-1011 at 30,000 feet, with three hours to go before landing in Los Angeles, and the "low battery" light comes on. If your computer were a Model 100 or 200, you would simply pop out the exhausted AA cells and load in four fresh ones. But you wanted an eighty-character display and a disk drive, so on this trip you brought your Tandy 600. It gets its power from a built-in nickel-cadmium (nicad) battery, and the only thing Radio Shack sells to power it externally is an AC adapter. The nicad is sealed up inside the computer. Is there any way you can keep on typing?

This article discusses the power requirements of the Tandy 600, and tells you how an investment of about \$3.50 lets you make an external alkaline-cell power supply for the computer. When the "low battery" light comes on, you will simply connect the external battery pack. If the external battery runs down, causing the "low battery" light to come on again, you will be able to replace the D cells with fresh ones, and keep on typing.

Power Consumption

When connected to its AC adapter, the Tandy 600 is rated to consume 1.5 amperes (A) at 8 volts (V); multiplying the two figures yields a power consumption of 12 watts (W). This is an order of magnitude higher than the rated AC-adapter power consumption of the 100 (1.1W) or the 200 (1.2W).

(Carl Oppedahl is a lawyer specializing in technological litigation. He is the author of Inside the TRS-80 Model 100.)

What's Inside

Inside the 600 is a nicad battery composed of four cells; the no-load output is about 5.25 V. It is positioned at the left rear of the computer and provides a nice counterbalance to the mass of the disk drive.

The AC adapter for the Tandy 600 has no catalog number, so it is not clear how you could buy an extra adapter to have one at the office and one at home. It is rated to produce 8 V at 1.5 A and contains a large transformer, a full-wave bridge rectifier and a 4700-uF 16-V electrolytic capacitor. A 4-A fuse is in series with the DC output, which terminates in a 5.5-millimeter diameter coaxial power plug with positive current at the outer conductor.

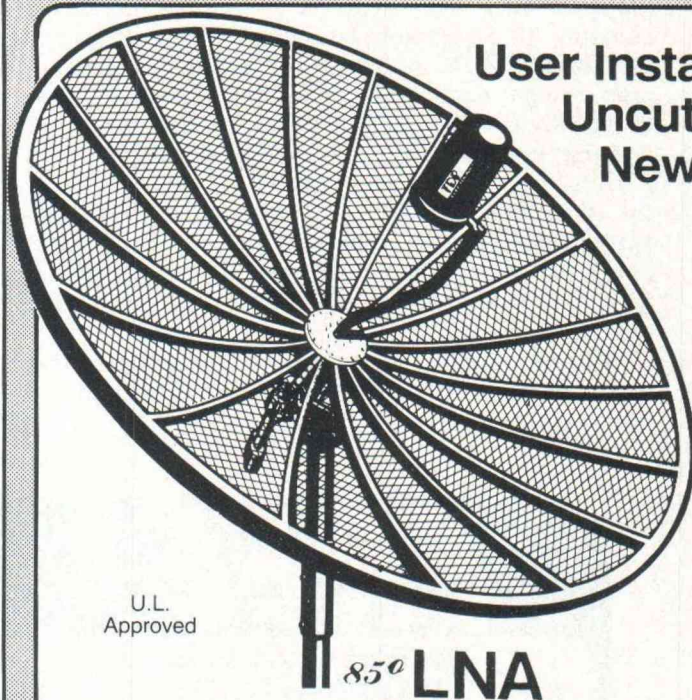
Designing a Substitute

Designing an external supply for the Model 100 or 200 is quite easy — since the adapter jack is labeled for 6 volts, a 6-volt lantern battery or 6-volt, lead-acid gel-cell will serve perfectly. But the adapter jack on the Tandy 600 is labeled 8 volts, and no common, commercially available battery produces 8 volts. What's worse is that no combination of 1.5-volt AA, C, or D cells can add up to 8 volts — the closest you can get is 7.5V (5 cells) or 9V (6 cells). To decide which to use, I measured the actual power used by the Tandy 600 under various conditions.

Disconnected from the computer, the AC adapter produces more than 11 volts. With various test leads connected, I found that when the computer is turned off, charging of the nicads can consume 4.2 watts (0.5A at a time when the power supply produces 8.4V). Turning the power

Discover a Whole New World
of TV Entertainment!

SATELLITE SYSTEM



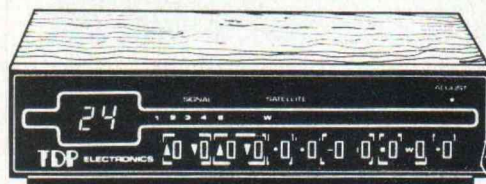
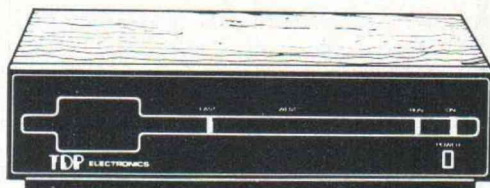
User Installable! Over 100 Channels!
Uncut Movies! Specials! Sports!
News! Weather! Music Videos!

Entire
System

\$1519

- Totally Automatic System With 8½ -Ft. Dish, Receiver and Wireless Remote Control
- Comes With All Accessories Needed for Complete Do-it-Yourself Installation
- Descrambler Ready

Handy Wireless Remote Lets You Change Channels, Control Volume and Fine-Tune



Now you can enjoy it all! 24-hour non-stop entertainment that you simply can't get from local TV, cable, and even video tape! You'll see the hottest stars, recent uncut and uninterrupted movies, the ultimate in sports coverage—and the best part is, there are no monthly fees to pay. You'll see exciting family-oriented pro-

grams, plus music videos, FM broadcasts and movies with provision for stereo, and 24-hour news coverage on both TV and radio. It all adds up to out-of-this-world TV viewing at a down-to-earth price. Come in today and discover how easy it is to enjoy satellite programming! Includes complete installation manual.

(817) 825-4027

NOCONA ELECTRONICS • Box 593 • Nocona, TX 76255

UPS Shipping

on increases the power drain by 0.8 watts (to 0.6A at 8.3V). Disk operation really eats up the power, drawing another 4.25 watts (a total of 1.25A at 7.4V). Modem and printer operations can take a little more power still.

What this means is that five D cells, producing 7.5V, would barely squeak by as a substitute for the AC adapter. A battery pack of six cells, yielding nine volts, does a pretty good job.

Making the External Supply

The shopping list is pretty short. See Table 1.

Table 1
Parts list for external battery pack

Item	Cat. No.	Price
Battery Holder — 2 D cells	270- 386	1.19
Battery Holder — 4 D cells	270- 389	1.29
Coaxial DC power plug	274-1569	.99
Total		3.47

The battery holders have holes drilled for mounting, and I used screws to secure them to a piece of wood. Each holder has two wires coming out, one red (positive) and one black (negative). The two-cell holder produces 3 volts, the four-cell holder produces 6 volts, and the two in series produce 9 volts. I soldered the red wire of one holder to the black wire of another; the remaining two wires I connected to a length of lamp cord. The cord I had used one copper conductor and one silver, and I connected the copper conductor to the red wire and the silver conductor to the black wire. (If the lamp cord you use is not color-coded,

the insulation may have molded-in ridges to keep track of the conductors.) At the power connector, I connected the copper conductor to the outer (barrel) contact and the silver conductor to the inner contact. The plug has a plastic body which should be slipped onto the cord before soldering the cord to the plug.

It is of crucial importance that the polarity at the plug be correct. If there is any possibility you have gotten it backwards, put D cells in the holders and use a voltmeter to test the polarity; it should be the same as that of the AC adapter.

Using the Adapter

The main reason for hooking up the external D cells is to allow continued computer function. But assuming the "low battery" light was what prompted connecting the external supply, then the nearly-drained internal nicads are going to draw lots of power from the D cells just for charging. In other words, when the D cells are exhausted, the nicads may have charged up enough to allow a little more computing time. In any event, if you are going to carry around six D cells, you might wish to carry twelve to allow a change after the first six have been drained.

How long can you expect a set of D cells to last? Duracell Inc. tells me that with a constant current load on one ampere, their D cells could be expected to last perhaps five hours. Assuming minimal disk use, my calculations suggest a life of almost 10 hours. Disk access obviously uses up the cells much faster.

With this external power supply for the Tandy 600, you need not be so concerned about being away from AC outlets.

PCM

DON'T MISS THE LATEST ADVANCE IN MODEL 100/TANDY 200 SOFTWARE

X-TEL™ by the creators of TELECOMMUTER™

- Use TEXT while online in TELCOM: Download letters and upload a reply in one terminal session
- Exchange .CO, .BA and .DO files with other computers, bulletin boards, and information utilities
- Standard XMODEM error-checking protocol
- List and Kill Files online in TELCOM
- Compatible with Telecommuter and other desktop communications software supporting XMODEM

Specify Radio Shack Model 100 or Tandy 200 computer
Price: \$50 on MS-DOS diskette or cassette
MasterCard and Visa accepted.
Please call or write for further information.

Sigea Systems, Inc.

19 Pelham Road
Weston, MA 02193
(617) 647-1099

SO SIMPLE

FREE!

WORD PROCESSING TRAINING GUIDE for IBM-PC and TRS-80

Become an expert in the advanced word processing techniques in just a few hours. Develop skills that can be applied to any of the best selling word processors, no matter which system you decide to buy. Includes program disk so you learn quickly by doing. The training guide and training program disk (a \$50.00 value) will be sent to you absolutely free, with no obligation, upon receipt of this filled out coupon. Offer good for a limited time.

LeScript Word Processing Systems

P.O. Box 361136, Melbourne, FL 32936
(or call 1-305-259-9397)

Type of computer system:

- | | |
|---|--|
| <input type="checkbox"/> IBM-PC/XT/AT/PC jr | <input type="checkbox"/> TANDY-1000 |
| <input type="checkbox"/> TANDY-2000 | <input type="checkbox"/> TRS-80 1/3/4/4P |
| <input type="checkbox"/> TANDY-1200 | <input type="checkbox"/> TRS-80 2/12/16 |

Name _____

Address _____

City _____

State _____

Zip _____

Telephone _____

*(Computer names are tradenames and/or trademarks of their respective manufacturers.)

PCMfest Reporter

Vol. 1, No. 1
October 1985

Prospect, Kentucky
© Falsoft, Inc. All Rights Reserved

Editor: Danny Humphress
Writer: Jeffrey S. Parker

Record Crowd Attends Dual Festival

Against a backdrop of the RAINBOWfest Princeton, the first PCMfest was held October 11-12 at the Hyatt Regency Hotel, Princeton, New Jersey. RAINBOWfest is a show geared solely for the Tandy Color Computer, and nearly 13,000 people attended this huge show. PCMfest ran concurrently in the same location, and because of the tremendous interest from the Color Computer fans, no really accurate count of attendees is possible.

PCMfest was designed as a show where readers of Falsoft's PCM magazine would have a chance to meet some of the editorial staff of the magazine. Visitors could also attend informative free seminars on all aspects of personal computing and have a chance to make a hands-on inspection of new introductions in the MS-DOS family of hardware and software products.

Visible at the show were some well-known names to PCM readers, names such as Danny Humphress, Lonnie Falk, Jutta Kapfhammer and Dan Downard. The keynote speaker for the Community Breakfast was Bill Barden, who has written some 30 books on computers, and is one of the most widely read computer authors in the country. Bill spoke of future predictions, particularly in light of Tandy's entry into the MS-DOS market. "I predict that Radio Shack will be around forever," he said, and went into a discussion about the merits of desktop computers. Attendees had a chance to see many products demonstrated, such as The Small Computer Company's *File Pro 16*, or Sigea System's *Telecommuter* software. Osicom showed off the Rose Hard Drive, and Tandy went all out to market their laptop and new MS-DOS machines at special show prices.

Howard Medical Cuts Monitor Prices

Howard Medical Co., of Chicago, cut the price of the Zenith 131 monitor to \$168 for the show. This 13-inch monitor has 240 X 200 resolution, a built-in speaker, an RGB and a composite jack, and is backed by a 30-day, money back warrantee. In addition, Howard Medical was selling the Zenith 123A 12-inch monochrome green screen monitor for just \$67.50.

Computer Discount of New Jersey

Computer Discount, a division of the Crowley Corporation, offered a wide range of equipment at huge discounts at PCMfest. On display was the Rose-20, a 20.15 megabyte hard drive for the Tandy 1000. Available also was the PBJ MFB with DMA controller chip, clock/calendar, serial port, 512K of RAM memory and more. Computer discount offered a stuffed Tandy 1000 with 640K, second drive, monitor, printer, DOS and BASIC at the astounding price of \$1,999, a savings of nearly \$1,000. Also on display was the AT&T PC 6300, a different MS-DOS alternative to the Tandy 1000.

Hard Drive Specialist Displays TanPak, Hard Drives

Extra drives, memory upgrades, the TanPak multi-function board, and many hard disk subsystems for the Tandy 1000 at bargain prices were the order of the day for HDS. This exhibit was usually several rows deep in people looking for good quality, low cost ways to add a hard drive, serial ports, clock/calendars and the like to their Tandy 1000s. HDS was a major supporter of PCMfest at the show, providing show goers with an opportunity to get a thorough inspection of the equipment before purchasing.

Sigea Systems Introduces *Telecommuter* Software

Sigea Systems, owned and operated by Harry and Ellen Brawley of Weston, Massachusetts, was on hand to demonstrate their powerful new *Telecommuter* software. This package is designed to transfer files between a portable computer, such as the Model 100 or 200, and a desktop, like the Tandy 1000, 1200, 2000, etc. The logo of Sigea Systems is "So Simple," and *Telecommuter* is one of the easiest to learn and simplest to operate programs ever created for the MS-DOS family of computers.

More than just a file transfer utility, *Telecommuter* will take a file from a laptop and transport it intact to a desktop, where it can be edited. The process can also be reversed.

Sigea Systems has also built a full-function, easy-to-operate word processor into *Telecommuter* that makes some word processors look like tinker toys. Sigea Systems gives you editing and formatting, and as their display indicated, all the bells and whistles.

PCM Booth Kept Busy

Many well known names from the PCMeditorial staff were present to answer questions and help people wanting to purchase a Falsoft publication, or just say hello to someone previously only known over the telephone. Despite the tremendous crowd, the staff of PCM managed to keep things running smoothly.

Computer Plus Sets Discount Trends

Computer Plus, of Littleton, Massachusetts, had many show specials available, including the Gemini SG-10 printer for under \$245. This was just a sample of some real bargains available at this exhibit, as Computer Plus is an authorized Radio Shack dealer, but discounts its items 10-20 percent and more. Imagine a Tandy 1000 with a monitor for just \$859!

The Small Computer Company Displays *filePro 16*

filePro 16 is a multifunction, multipurpose program developed for single and multi-user computers by the same company that brought *Profile* to the world of TRS-80 computers.

filePro 16 and *filePro 16 Plus* are database management systems that also incorporate an applications developer. This system is so

flexible and powerful, particularly when coupled with the applications developer in *filePro 16 Plus*, that there is very little this program could not handle. *filePro 16* also interfaces with major programs like *Multiplan* and any major word processing system, and will run on a wide field of MS-DOS, PC-DOS, and even Unix/

Zenix and CP/M machines.

Because of its power and flexibility, and its ability to be conformed, configured and designed to the user's specific needs, this program is a must for the business environment in need of high powered data management and total flexibility.

Informative Seminars Highlight PCMfest

The seminar topics were wide, varied and quite impressive. Nearly every seminar drew a capacity crowd in the seminar rooms. The topics covered many of the important areas of computing that concern all users of portables, desktops, MS-DOS and compatible machines.

Danny Humphress, managing editor of PCM, gave a talk on Tandy's new intra-

office communications system, *ViaNet*, which is compatible with all of Tandy's MS-DOS and Zenix computers. Easy to learn and use, *ViaNet* doesn't make the rest of the software in a system obsolete. This is Tandy's excellent new entry into the networking field.

Other topics covered law and portable computers, database management and data communications; Bill

Barden gave a compelling talk on computer languages, a fantastical journey into the future of portable computing, and Harry Brawley presented a detailed discussion of the implications of new software that allows use of the same commands for portables and desk top computers, and what this new development will mean to the world of laptop computing. □

cheapware® Games

for the Tandy™1000, 1200, 3000 and compatibles

Arcade Poker: You've played it in bars, in video arcades and in Vegas. With this program you can either play poker for points to beat your high scores or play against others to beat their high scores. **\$15.00**

Draw Poker: Includes Arcade Poker, play against the Dealer and two games to play against others, either for fun or for money. **\$35.00**

Poker Odds Calculator: Compute exact odds of holding any poker hand at the end of the second draw with this program. **\$25.00**

Gramarcy® —the Word Game: This is the same game you will see in your daily paper. You must find all words in a given word using each letter only once. Utilities are included to update data base. **\$35.00**

Price includes first class postage. VISA/MasterCard accepted.

Information/Orders call:

Robert L. Nicolai 800-NICOLAI (642-6524)
4038 N. Ninth Street or
St. Louis, MO 63147 314-621-7618 in Missouri

Tandy is a trademark of Tandy Corporation



EDUCATIONAL/ENTERTAINMENT PROGRAMS FOR THE TANDY 1000

Disk 1	Number Fun! (age 4-8)	\$15.00
Disk 2	Kid's Adventures (age 4-10)	\$15.00
Disk 3	Picture Puzzles (256K) (Child-Adult)	\$15.00
Disk 4	Learning Graphics in Basic	\$15.00
Disk 5	Games (age 4-99)	\$15.00

Price includes postage and handling in the U.S.
Foreign orders add \$2.50.

Send \$1.00 for list & description of programs available.
Refundable with first purchase.

Send check or
money order to:

(U.S. funds only)

SODA POP SOFTWARE
P.O. BOX 653
KENOSHA, WI. 53141

'DB11'

A Towering Database System

By Robert D. Covington

In January, we introduced Robert Covington's DB11 database management program and discussed the concepts and basic modules of this system. This month, you'll find BASIC listings for the complete program as well as a discussion on the report generator and the other, more advanced, features of DB11.

Because of program enhancements, the specific instructions for loading and saving the first three modules which appeared in last month's Subroutine City have been changed. Instead, follow the instructions given in this article. The instructions presented for using DB11 have not changed, however.

Last month, I described two modules for the *DB11* database system which allowed a database to be created and allowed data to be entered and edited in that database.

This month, the remaining parts of *DB11* will be discussed. These parts include the report generator, the data manipulation functions, and the rest of the database format functions.

Installation of Programs

Programs 1 through 5 contain the entire *DB11* database system. Each of these programs should be entered separately in BASIC and saved with the instruction `SAVE "PROGn.BAS"`, where 'n' is the program number (1 through 5).

Once the programs have been entered, follow the instructions given in Figure 1. All of these instructions should be entered in BASIC's command mode exactly as they are printed. Before starting the instructions, however, make at least one backup of your original programs. That way, if you make a mistake, you can start again with the backup. These instructions assume there is plenty of room on the disk. When all instructions have been completed, *DB11.BAS* and *PROG1.BAS* through *PROG5.BAS* can be deleted from the database disk.

(Bob Covington has been a computer programmer and consultant for the past six years, most recently focusing his attention on both the Model 100 and the 2000. He is also a technical writer and editor. Bob can be contacted at P.O. Box 37007, St. Louis, MO 63141.)

Figure 1

Instructions for creating MAIN.BAS

```
LOAD "PROG1.BAS"
MERGE "PROG2.BAS"
DELETE 112-113
DELETE 116-118
DELETE 156-157
DELETE 165
DELETE 250-295
DELETE 50800-50820
DELETE 52400-52595
DELETE 53000-53025
DELETE 53300-53540
GOTO 65000
```

Instructions for creating DBFORM.BAS

```
LOAD "PROG1.BAS"
MERGE "PROG3.BAS"
DELETE 112-113
DELETE 116-118
DELETE 156-157
DELETE 165
DELETE 250-295
DELETE 50100-50795
DELETE 51100-51105
DELETE 52200-52305
DELETE 53300-53540
DELETE 54000-54205
GOTO 65000
```

Instructions for creating DATAMAN.BAS

```
LOAD "PROG1.BAS"
MERGE "PROG4.BAS"
DELETE 112-113
DELETE 116-118
DELETE 156-157
DELETE 165
DELETE 200-295
DELETE 50800-50820
DELETE 51100-51105
DELETE 52200-52595
DELETE 53000-53025
DELETE 53300-53540
```



```
DELETE 54000-54205
GOTO 65000
```

```
Instructions for creating PRINT.BAS
LOAD "PROG1.BAS"
MERGE "PROG5.BAS"
DELETE 200-235
DELETE 52200-52595
DELETE 53000-53025
DELETE 54000-54205
GOTO 65000
```

After the BASIC programs have been created, exit to DOS and type the text in Figure 2 at the MS-DOS prompt. This will create a batch file for DOS that loads the *DB11* program into memory.

The next step is to create a default CDBASE.PRT file. While at the MS-DOS prompt, type the following:

```
COPY CON CDBASE.PTR
00
00
0
0
```

^Z (CTRL-Z followed by ENTER)

Once all of the above steps have been completed, type *DB11* and press ENTER. The database system will be loaded into memory and initialized.

Because of the size of *DB11*, it is usually wise to use a minimum MS-DOS system disk for storing the database system. To create such a disk, format a blank disk with the *FORMAT /S* command and copy *BASIC.EXE* and all the database files to the new disk. Doing this will give the *DB11* system enough room on the disk for its internal files.

Database Format Functions

Last month we discussed only the minimum functions required to create and initialize a database. This month, six additional functions are discussed to simplify manipulating databases and their sub-databases.

The first new function, "Change Current Key Database," allows another database on the system to be used. When this option is selected, the program prompts for the drive to search for a new database. The drive letter that you enter here is searched by this function for any defined *DB11* databases. If the program finds one or more databases on the drive, the program allows you to select the database you want. If no database is found on the disk, the program generates an error and returns to the menu. This function is particularly useful when more than one database is stored on a drive (especially a hard drive).

To facilitate removing databases and

TANDY[®] COMPUTERS

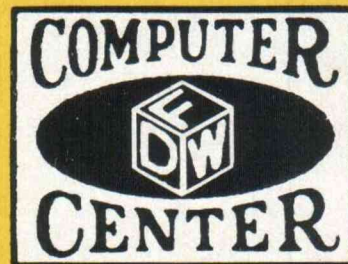
WITH THE
MANUFACTURERS
WARRANTY

ALWAYS AT "S"



- ☒ DISK DRIVES
- ☒ PRINTERS
- ☒ MODEMS
- ☒ SOFTWARE
- ☒ ALL RADIO SHACK PRODUCTS

☒ CHECK OUR SALE PRICES --- CALL -----



DFW/COMPUTER CENTER

Customer Service 817-481-SALE (9 am-5 pm CT)

SALE" PRICES



sub-databases from the system, "Kill Sub-database" and "Delete Current Database" have been added. The Kill Sub-database function allows one of the sub-databases defined in the current database to be erased. The Delete Current Database function erases the current database's field definitions, reports, sub-databases and data. Both of these functions are quite dangerous and generally irreversible. Use them with care!

For changing the characteristics of a database, the "Change Field Name" and "Extend Maximum Record Limit" functions have been added. The Change Field Name function allows the text associated with a database field to be altered. This function is most commonly used to correct typing and spelling errors in the field name that was introduced when the database was defined. The Extend Maximum Record Limit function changes the number of records that *DBII* allows you to store in the currently selected database. While this function's name is Extend Maximum Record Limit, the record limit can just as easily be decreased.

The last database format function, "List Database Characteristics," generates a comprehensive listing of information on the current database. The output from this function is quite useful for planning new sub-databases or creating new databases from old ones. For permanent records, this function allows the database information to be listed to a printer.

Data Manipulation Functions

The "Data Manipulation Functions" module contains four major functions for organizing the data in a database beyond that which can be performed by the edit functions.

Probably the most used function in this module is the "Sort Database" function which sorts the current database's data into ascending order. When the sort function is selected, the program asks for the sort priorities for each field in the current database. Fields with low sort priorities are used to order a database before fields with higher sort priorities. For example, in last month's sample mailing list database, if ZIP code was given a sort priority of one and name was given a sort priority of two, the sort function would first sort by ZIP code then by name if two or more ZIP codes were identical.

Once all the fields in the database have been assigned a sort priority, the program proceeds to sort the database.

Do You Want A Real Warranty. . .

or will you settle for a vague promise?

Since 1977, we have operated a Radio Shack Authorized Sales Center (ASC), in the city of Grapevine, Texas. This will be significant for you only if (a) you want a warranty which you can exercise easily, if necessary and (b) the reassurance of our long term stability and business ethics.

When you buy a Tandy/Radio Shack product from us the Tandy/Radio Shack Warranty will accompany it and the warranty and service will be available to you, anywhere in the U.S.

We will not install any foreign parts which might, and probably will, have an adverse affect upon your warranty. We will assist you in obtaining local support, should you experience difficulty, and will make an offer to buy-our-product-back* (hardware) if it dissatisfies you, within 30 days.

"We ship fast," has always been our motto and if anyone is offering you a lower price, we suggest you ask about the foreign parts (and warranty) before closing the deal. We will "NOT meet-or-beat" a lower price so you can be sure you have our best quote the first time, and our toll-free phone lines (outside Texas) make the quote easy for you to get, from 9 a.m. - 5 p.m. central time, Monday thru Friday.

(*For a small handling charge)

TOLL FREE 1-800-433-SALE

326 Main St. Grapevine, Texas 76051

TEXAS BUYERS ADD 5 1/8% SALES TAX

Unlike many databases, *DB11*'s sort does not require any additional overhead for sorting the database data. This means that you can sort a database that occupies an entire disk without needing an additional disk for the sort.

While the sort function is probably the most used function in this module, definitely the most powerful function is the "Transfer Database Data" function. This function allows data from one database to be transferred to another database. On the outside, this function might not seem very useful. At closer inspection, however, you will find this simple function adds one of the most useful functions that always seems to be omitted from many commercial database systems: the ability to delete or add fields to an already existing database.

When the Transfer Database Data function is invoked, the program asks you to select a source database. As you might expect, the data that is to be transferred comes from the source database. The destination database is always defined as the current database. In most cases, the destination database has been recently created and contains no data.

Once the source and destination databases have been defined, the program has the user define which fields in the source database should be trans-

ferred into the destination database. In this process, the program displays each destination database field one at a time and has the user either join it with a field in the source database or leave the destination field empty.

Two fields are allowed to be joined only if their general types are identical. For example, an alpha-only field can be transferred to an alphanumeric field but it cannot be transferred to a numeric field.

Once all the transfer definitions have been set, the program proceeds to transfer the database data between the joined fields of the two databases. When the data is transferred, it is appended to the end of the destination database. All "tag," "new record," "edit" and "delete" statuses are preserved in the transfer.

Since this function is rather complex and powerful, let's go through an example of adding a field to last month's mailing list database. The first step is to define a database with all the original database fields plus any additional field.

For our example, let's say we wish to add a field for the person's title. Since we are creating a new database, go to the Database Functions menu and select the Make Database option. When creating the database, make sure to use a database name that is different from the original mailing list database. If the

same name is used, the new database will overwrite the old one and all its data will be lost.

Once the new database with the additional field has been created, return to the Transfer Database Data function in the data manipulation module. The first question asked by this function is what drive contains the source database. For this example, let's say the old mailing list database is on Drive A. After entering the A, the program asks you to select the source database from the list of available databases on Drive A. To do this, press the up or down arrows until the name of the old mailing list database appears on the screen. When the desired database is found, press the space bar to select it. At this point, the program now knows that the source database is the one you just selected and the destination database is the one you just created (remember, all new databases are automatically defined as the current database).

After the transfer databases have been selected, the program asks you to select the source field that should be joined with the first destination field. In this example, the first destination field, NAME, should be joined with the source field NAME. To select the source field, press the up- and down-arrow keys until the NAME field appears on the screen. Press the space bar to join the two fields. If for some reason you did not want the NAME field to be transferred from the source to destination databases, you could then press ENTER to skip the current destination field.

Once the NAME field has been joined, repeat this process for the rest of the database fields. When you reach the new TITLE field, skip it since there is no source database field to join with it.

When all the field transfer parameters have been set, the program proceeds to transfer the source database field data to destination database fields. When

Figure 2

```
Instructions for creating startup batch file
COPY CON DB11.BAT

ECHO OFF
CLS
ECHO DB11 Database Management System
ECHO By Robert D. Covington
ECHO Copyright 1986 PCM Magazine
ECHO Loading database. Please wait.....

BASIC MAIN /F:6 /S:1024
^Z          (CTRL-Z followed by ENTER)
```

TANDY 200 24K EXPANSION MODULES \$99

8K MEMORY MODULE \$29

For your Model 100 & NEC PC-8201. Field proven (over 30,000 in use), factory tested and fully supported.

SIDESTAR \$399

128K RAM DISK CARTRIDGE

For the Starlet NEC PC-8401

SIDECAR 128K RAM CARTRIDGE \$259

Plugs into side of the NEC PC-8201



420 Constitution Ave.
Camarillo, CA 93010

Portable Thermal Printer

TTXpress MODEL 1280 \$99

Battery operated, 2.2 lbs., 8½" paper, 40 cps., parallel.

FREE SHIPPING, HANDLING & INS. FOR CONT. USA.

Orders shipped UPS next business day. 30-day satisfaction, money back guarantee. We accept Visa, M/C & Amer. Exp. Checks held 18 days.

(800) 732-5012

(805) 987-4788 (in Calif.)

**CALL FOR
LATEST PRICE**

completed, the new database will contain all of the data that was in the old one with the addition of one additional field (initially blank).

The last two functions that are implemented in the Data Manipulation module of *DBII* are responsible for deleting data from the current database. The Erase Database function causes all of the data stored in the database to be erased. After this function has been used, the database is in the same condition it was in just after it was created. In other words, even though the data in the database was erased, new data can be entered into the database at a later time. The actual database definitions are not altered.

The Organize Database function eliminates all deleted data below a certain level from the database. As you may recall, when records are deleted in *DBII*, they are not actually removed from the system. The deleted records are assigned the current delete level and logically made invisible to the system. By doing this, records that have been deleted can be retrieved at a later time if necessary.

Sometimes, however, you do not want to waste disk space keeping old deleted records. To eliminate this problem, the organize database function will actually remove the deleted data from the disk and recover the space it occupied for future records. Unlike some systems, organizing the database in *DBII* does not change the relative position of the records or require any additional disk space.

Report Functions

Probably the most important section of a database is its report generator. A report is a portable extension of the organizational capabilities of a database system. Without reports, we would need to have the computer and database system available any time we needed to find information that is stored in the database. If a database's report generator is not flexible enough to make a report the user needs, the entire database's productivity is greatly reduced.

The report generator in *DBII* is one of its most powerful features. *DBII* supports both single- or multi-line columnar reports (traditional reports) and multi-line block reports (mailing labels) which can contain some or all of the fields in a database. In addition, reports can be based on the find parameters and the "tag," "edit" and "new record" statuses.

Reports on *DBII* can be output to

any legal device or file in the system. To support this flexibility, the "Print Report" functions module contains four major functions for manipulating and defining output devices for reports.

Output Driver Functions

The most important output driver function, "Define Output Driver," creates a printer driver which interfaces the *DBII* Print Report function to an output device. When this function is invoked, the program asks for the driver name. This name can be any text string that distinguishes the output driver from others.

Next, the program asks for the device name. In most cases, the output driver will be LPT1: (line printer Port 1). In some cases, however, you may have printers connected to other printer ports (LPT2:, LPT3:, etc.) or to an RS-232 (COM1:). If the RS-232 is selected as the output port, use the format for the RS-232 parameters described in the BASIC manual under the OPEN~COM1:~ instruction.

In addition to devices, the device name can be replaced with a filename. This function is very convenient in interfacing data from *DBII* to many spreadsheet and word processing systems. If a question mark is used for the device name, the program will prompt for the device name each time the report is printed. This function was implemented to better facilitate printing reports to a disk file. If this function were omitted, you would have to define a printer driver for each file you wanted to output to. If text follows the question mark (example: ?LPT1:), the additional text is used as the default for the device name prompt asked before printing.

After the printer driver has been named and the output device has been selected, the program prompts for the control codes necessary to select various fonts or pitches on the output device. When the program prompts for a font code, it asks for the font's name and the decimal control codes required to select the desired font. If the device you are connecting to does not require any font control codes — a disk file or the screen (SCRN:), for example — configure a font that sends a useless character to the device or file. Usually, the best control code to use in this case is a NUL (decimal 0).

Once all the fonts have been defined, the program then has you choose which font you wish to use for printing. This font is stored as the current font and used whenever the printer driver is used.

To change the current font once the printer driver has been created, the "Font/Pitch Change" function can be used. As the name implies, this function allows a new font to be defined in the printer driver.

To change the current printer driver to a previously defined driver, the output driver selection function can be used. *DBII*'s ability to support a variety of printer drivers is very convenient when more than one printer is connected to a computer. This is especially true in a Vianet network where multiple printers are common. When a printer driver is selected, the current font is automatically set to the last font used by the printer driver.

The Report Generator

Once an output driver has been created and selected, reports can be generated and printed. To do this, *DBII* contains four functions for creating, printing and manipulating reports.

Probably the most important of these functions is the "Report Generator" function which is responsible for creating the format of a report. When this function is invoked, you are prompted for the report and the margin settings of the report page. When entering margin settings, make sure your printer can handle the page you have defined. This is especially true with wide carriage printers using 8½ by 11 sheets of paper. If you define a page that is larger than the paper size, you will probably end up printing on your roller.

Once the page size definitions have been set, the report generator enters a full-screen editor where the user can input the report definition. The full-screen editor used in the report generator is very similar to the editor in the Build Database function. Table 1 contains a listing of all the control codes that are available in the editor. The main difference between the two editors is the report generator's editor can manipulate an area of up to 255 characters by 24 rows instead of the normal 80 characters by 25 rows. The editor emulates a 255-character-wide screen by using the computer's 80 by 25 screen as a window on the actual editor page.

A report definition contains three major parts. The top part of the report usually contains header information, page number definitions, titles, report dates, etc. This information is usually only printed once on each page.

The middle part of the report contains the actual database record data. In a normal columnar report, each line in the middle section corresponds to a

record of information in the database. When the middle section is defined, one copy of the information is formatted for the entire page.

Then, the computer prompts for the area of the report that is to be repeated for each record of data. By doing this, the user does not have to enter report definitions for every line in a printed page of a report.

The end section of the report usually contains footer information, page

numbers, report names, etc.

Each of these three sections can contain three types of information: text descriptions, virtual fields and database fields.

Text descriptions are non-field data that are usually used to describe fields or reports. Text descriptions are usually used for column headings and report names.

Virtual fields are fields of information that are either generated by the

program when the report is printed or retrieved from the record status file. Examples of virtual fields are page number, line number, delete level, edit status, etc. Table 2 gives a complete listing of the virtual fields along with their descriptions. When virtual fields are defined, they should be treated just as if they were actual database fields.

Database fields, as the name implies, are fields of information that are stored on the current database. If you recall, when defining a sub-database, database fields are signified by a colon (:) followed by a string of periods. When a report is generated, however, database fields use slightly different field identifiers. A complete listing of these report field identifiers is given in your system's BASIC reference manual for the PRINT USING instruction. As you can see from this information, the identifiers are divided into two major groupings; text (string) fields and numeric fields.

Text fields are usually identified by two backslash characters (\). For example, if you wished to define a text field of 10 characters, you would type a backslash, eight spaces (or any other character) and another backslash. To define text fields that only contain one character, the exclamation point (!)

Table 1

Full screen editor commands for report generator

CTRL-D	Duplicate last character for rest of line.
CTRL-E	Erase current line and index lower lines up.
BACK SPACE	Backspace and delete.
HOME	Move cursor to upper-left corner.
END	Move cursor to lower-right corner.
CTRL-L	Insert line at current row.
ENTER	Move to beginning of next line.
CTRL-Q	Quit editing screen and return.
INSERT	Insert space at current cursor position.
DELETE	Delete character at current cursor position.
ESC	Erase (blank) current line.
Arrows	Move cursor in direction of arrow.
Shift Arrows	Move to extreme position in arrow direction.
PG UP and PG DN	Scan through field information (field help).
CTRL-I	Insert report field for current help field.
TAB	Same as CTRL-I.

FAST CASSETTE STORAGE

for Model 100 Text Files

* Save .DO files up to **THREE** times as fast

* Load text more than **TWICE** as fast

WHAT IS "FAST"?

FAST™ is a revolutionary concept in the way the Model 100 handles the storage and retrieval of text (.DO) files. FAST saves or loads text in one continuous stream of data — the same way BASIC programs are saved and loaded. The package (compatible with M/L programs) consists of two easy-to-load programs:

The **SAVER-BACKUP** program occupies approximately 900 bytes of RAM — small enough to always carry along, whether at home or on the road.

The **LOADER** program, which occupies 2K of memory and reserves about 1K during loading, makes loading text easier than ever. When loading, a unique **Search Mode** feature presents the name and length of each file encountered, and allows user to decide whether to load, or to skip on to the next file on the tape. And unlike the "Old Way," the program will warn the user before loading if insufficient memory is present, and exactly how much more will need to be freed. Error-checking has been improved (no more dropping of 256-byte chunks of text), and now you can even load flawed tape files — which are usually still usable, and maybe even crucial.

The Facts: The recording time to save a 5K file is now only 39 sec (vs 106 sec). A 10K file takes 1.1 min (vs 3.4 min). But there's more . . .

OTHER FEATURES:

- **Backup** all your irreplaceable Text Files (including ADRS and NOTE) in one quick, convenient operation. This feature alone is worth the price.
- **Verify** cassette saves before erasing files from memory.
- **Recover** data "lost" after a Cold Start.
- **Store** three times as much text on each cassette (500K per 60' cassette).
- **Access** cassette files three times as fast.

This totally NEW concept is from MinSof™ — which is committed exclusively to developing software for laptop portables.

Organization, Dealer, and all other inquiries: Send stamped, self-addressed envelope for quickest response.

\$19.95

Check or Money Order
(Please include ZIP code)

MinSof, Inc.
Box 1153
Minneapolis, MN
55440

Virtual field descriptions

Table 2

Virtual field name	Type	Length	Description
*Page Number	Integer	4	Current report page number
*Record Number	Integer	5	Record number
*Line Number	Integer	5	Count of printed records
*Delete Level	Integer	2	Delete level of record
Tag Status	Alpha/#	1	Asterisk () if "tag" set for record
Edit Status	Alpha/#	1	Asterisk () if "edit" status set
New Record Status	Alpha/#	1	Asterisk () if "new" record
*Report Time	Time	8	Time report started printing
*Report Date	Alpha/#	10	Date report started printing

character is used.

To simplify the entry of text fields, the full screen editor in the report generator will automatically insert a field identifier by pressing the tab key or CTRL-I. To select the field to insert, press the page up or down keys until the desired database field description is displayed on the last line of the screen. This feature is very convenient for entering long fields which are quite cumbersome to enter manually.

While most text report fields are the same length as their database field counterparts, the field lengths can differ. If a report field is smaller than the data that is going into it, the extra characters are truncated from the right side of the string. This feature is useful when report fields are too large to fit on the defined page.

Numeric field definitions use the number character (#) to signify digits in the field. For example, if you wished to define an integer with five digits as a report field, five number characters are used (#####). If the field is not an integer but a floating point number (a number with a decimal point), place a period where you want the decimal point to be. For example, to define a monetary field, the field identifier #####.## could be used. If a number that is going into this field contains more digits to the right of the decimal point, the value is rounded (34.567 is printed as 34.57 in our example). To add commas every three digits, place a comma in the identifier every three digits (##,###,###.##).

In addition to the special identifiers just described, about six additional field identifiers are described in the BASIC manual. I would suggest that you take a look at the PRINT USING function's identifiers so you can make the most of the report generator.

Figure 3 shows a sample columnar report definition for our demonstration mailing list database. The first line of this report definition contains the virtual fields of page number and report date. The fifth line of the report contains the repetitive field block that contains that actual database field data. The lines 1 through 4 and Line 6 are defined as header and footer text respectively.

Once the report has been defined, the program asks the user to define the start and end of the repeat block. This repeat block is the area in the report that is repeated for each record until the end of the page is reached. In most cases, this block contains only field definitions, but sometimes also includes text that will be repeated each time the block is repeated. This feature allows lines to be inserted between record blocks, double spacing of reports, etc.

The last step in generating a report involves joining a report field with a database field. In this process, the program displays one of the fields defined in the report. The computer then asks which database field describes the report field. Just like with most of the other sections in the database, the database field is selected by pressing the space bar after the field has been found

using the up and down arrows. If the program mistakes a group of text in the report definition for a report field, the report field can be skipped by pressing ENTER. This happens quite often when number signs (#) are used to abbreviate report headings.

Once the report fields have been joined to a database field, the report has been generated and is selected as the current report.

Other Report Functions

In most databases, there are at least two reports defined for a single database. To facilitate multiple reports, DBII's "Change Report" function allows other report definitions in the database to be selected and used for printing.

If a report is not going to be needed anymore, the "Kill Report" function will erase a report from the database system. If the currently selected report is deleted, remember that a new report needs to be selected or created.

Once a report and output driver have been defined and selected, the Print Report function can be used to print a report defined by the report generator. When this function is invoked, the program asks if you wish to print by the find parameters or by the "tag," "new record" or "edit flags." If you respond with a 'Y' to any of these prompts, only the records that match the selected qualification will be printed. For example, if you print by find and tag parameters, only the records that meet both requirements will be printed.

After these questions, the program asks for the access level of the report and asks if you wish to pause between pages (for single-sheet paper or output to the screen). Just like in the Edit/View Database function, increasing the access level allows records that have been deleted with delete levels lower than or equal to the access level to be printed. Once all of these questions have been asked, the program proceeds to print the report.

The find parameters used in the Print Report function are defined by the "Set

Figure 3
Sample Column Report Definition

```

Page Number: #####    Report Date:\-----\
-----
Name                Address                City                St  ZIP
-----
\-----\ \-----\ \-----\ \ \ #####
-----

```


Find Parameters" function. When this function is invoked, the program asks for the field to be searched and the text to search for. The program then asks which of the five relations should be used.

The first three relations describe three types of equivalent operations. The first, "absolute equivalent," is true when the search data and the database data are identical in every way. The second operation, "left part equivalent," is true when the search data is equivalent to the left part of the database data. For example, if the search data for this relation was an 'A', any database data beginning with an 'A' would be considered equivalent. The last operation, "instring equivalent," is true when the search data is found somewhere in the database data. For example, if the search data for this relation was 'B', any database data with a 'B' in it would be considered equivalent. These three equivalence operations are only effective when alphanumeric, alpha-only or time fields are searched. With numeric or date comparisons, any three of these operations will yield an absolute equivalent operation.

The last two operations, less-than and greater-than, as the name implies, are true when the search data is less than or greater than the database data. Both of these relations are based on the entire string.

Once the relation has been set, the above processes can be repeated for multi-field searches.

Help Functions

Throughout *DB11*, support for help files has been added to aid the user in

using the database without additional manuals. All of the help files available in *DB11* are ASCII text files stored on the default drive with the extension .HLP. Table 3 lists the help filenames along with a description of where the help file is made available.

When a help function is invoked, the subroutine at Line 50900 attempts to load in the selected help file and print it on the screen. If the help file is not available, a warning message is printed and control returns back to the area where the help was invoked.

If the help file is available, the text file is printed on the screen until a '\ ' character is found in the file or the end of the screen is reached. At that point, you are prompted to press any key to continue. The '\ ' character allows some formatting control of the help files by letting you control where the pages break.

For a help file to be printed correctly on the screen, a carriage return must be present on every line (every 80 characters) of the file.

Future Enhancements

As you look through the code of *DB11*, you will probably see that a lot of organization has gone into its general structure. Variables, subroutines and files are given consistent definitions throughout the program.

This structure was not added just to make the program easy to read. The primary reason for the consistency and the extended use of subroutines is to add an easy upgrade path to future systems.

For example, *DB2*, *DB11*'s bigger brother, replaces the current random

block disk I/O routines with ISAM file routines. Adding these file routines only requires replacing the subroutines at lines 50200 and 50300, and making a few alterations to the Sort Database, Organize Database, and Print Report functions. That's it!

Because of the flexibility of expansions with *DB11*, many new functions can be added relatively easily. Some of these new functions are *Lotus 1-2-3* interfaces, math functions for the report generator, ISAM file structures, binary tree indexes on all fields for nearly instantaneous searches and sorts, relational capabilities, more functions for the Edit/View mode, remote terminal access to the database, password protection and much more.

The amount of extensions to *DB11* that I publish, however, is proportional to your interest. If you are interested in seeing some of these added functions, drop me or PCM a letter expressing your interest.

End Notes

In January, PCM started offering programs that were published in their magazine on disk through a service called PCM on Disk. As you would suspect, all of *DB11* will be published in the February PCM on Disk. In addition to the actual program, help files, documentation files and technical data will also be included on the disk.

In next month's Subroutine City, I will describe how to compile *DB11*. Since *DB11* primarily uses integer variables, most functions increase in speed by a factor of at least four. The only exception to this is disk intensive operations (like sorts) which are limited by the speed of the storage device.

This installment ends this two-part series on the *DB11* database system. I hope that you find it as useful as I have in the past few years for handling light database applications. If you have any questions about *DB11*, MS-DOS, Xenix or the Tandy 1000 through 3000, please drop me a note and I will try to help in any way I can. Please include a self-addressed stamped envelope for the reply. □

Help file descriptions

Table 3

Help filename	Program called from	Program area where available
MAIN.HLP	MAIN.BAS	Main menu
EDIT.HLP	MAIN.BAS	H key in Edit/View function
DBFORM.HLP	DBFORM.BAS	Main menu
DATAMAN.HLP	DATAMAN.BAS	Main menu
PRINT.HLP	PRINT.BAS	Main menu

Listing 1:

```

1 *****
2 *                                     DB11 Database System *
3 *                                     by Robert D. Covington *
4 *                                     Copyright 1986 PCM Magazine *
5 *****
10 DEFINT A-Z:KEY OFF:WIDTH LPRINT 255

```

Continued on pg. 82

Grow up!

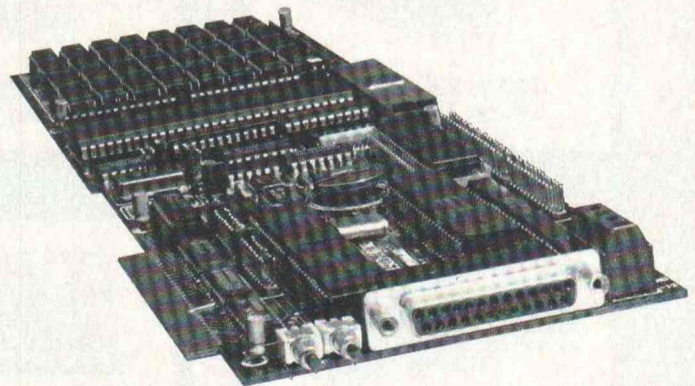
Wondering where to grow?

Why not get IBM compatibility with Tandy's 1200, or the new 3000, and take your OS-9 programs with you?

TLM's PC68K puts the 68000/68010 processor and Microware's OS-9 together in any IBM, or compatible, PC/XT/AT system, providing the best of both worlds with fully concurrent PC/68000 operation.

- Powerful languages such as BASIC09, PASCAL and C
- DYNACALC for spread sheet processing
- The power of the 68000/68010
- The price/performance and flexibility of the Tandy 1200 and 3000

TLM's PC68K... A great family to grow up with!



CLOCK SPEED

8/10/12.5 MHZ
Without wait states

HARDWARE

2 Parallel Ports
3-9 Serial Ports
68881 Floating point co-processor (optional)
Memory mapped dual port bus for high speed DMA transfers
Real time clock with battery backup
Up to 16 boards can run concurrently from one AT/XT/PC bus

MEMORY

1-2 MB OF RAM
(1 MB on board standard)
8K-32K of ROM
2K-8K of battery backed RAM

MULTI-USER

3-9 users (3 standard)
High speed "Tick" timer with 10ms accuracy for efficient multi-user time slicing.

OPERATING SYSTEMS

OS-9 (Multi-user UNIX look-alike)
C/PM-68K (popular single user OS)
Both support concurrent access to PC DOS

SOFTWARE

Software selectable OS environment
IBM Color/Mono Graphics Support
Local and Global disk caching for maximum speed and efficient data transfer

COMMUNICATIONS

Powerful asynchronous/synchronous data communications capability
Bisynchronous SNA 3270, X.25, etc.

MISCELLANEOUS FEATURES

Remote and Local reset (hardware or software selectable)
Remote and Local Non-maskable Interrupts (NMI)
Stand alone hardware capability (without AT/XT/PC support)

IBM is a registered trademark of International Business Machines.
DYNACALC is a registered trademark of Computer Systems Center.
OS-9 is a registered trademark of Microware Systems Corp.
CPM 68K is a registered trademark of Digital Research Corp.
MK68000 MK68010 are registered trademarks of Motorola.
UNIX is a registered trademark of AT&T.



From Computer Plus to YOU...

PLUS after PLUS after PLUS



Tandy 200 24K \$649
Model 100 24K \$425



Tandy 1200 HD \$1599
10 Meg. 256K



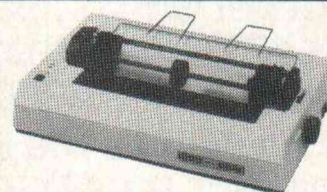
Tandy 1000 \$710
Tandy 1000 HD \$1539



Smith Corona L-1000
\$259



Color Computer II
w/64K Ext. Basic \$179



DMP105 \$160

BIG SAVINGS ON A FULL COMPLEMENT OF RADIO SHACK COMPUTER PRODUCTS

COMPUTERS

Tandy 1000 1 Drive 128K	710.00
Tandy 1000 HD 10 Meg. 256K	1539.00
Tandy 1200 10 Meg. 256K	1599.00
Tandy 2000 2 Drive 256K	1295.00
Tandy 2000 10 Meg. 256K	2025.00
Model IVD 64K with Deskmate	889.00

PRINTERS

Radio Shack DMP-130	269.00
Radio Shack DMP-430	660.00
Radio Shack DWP-220 Daisy Wheel	469.00
Radio Shack TRP-100 Portable	229.00
Silver Reed EXP-550 Daisy Wheel	400.00
Star SG-10	245.00
Star SG-15	410.00
Star SD-10	365.00
Panasonic P-1091	259.00
Toshiba 351	1175.00
CITOH Prowriter 8510AP + NLQ	345.00
Okidata 192	375.00
Epson LX-80	245.00
Epson FX-85	369.00
Epson HS-80 Portable Ink Jet	339.00

Tandy 1000/1200 ACCESSORIES

Tandy 1000 Disk Drive Kit	159.00
Tandy 1000 10 Meg. Hard Drive	579.00
Hard Drive Controller Board	249.00
256K Ram Board (inc. 128K & DMA)	199.00
512K Ram Board (includes 128K)	169.00
256K Memory Plus Expansion Brd.	249.00
PBJ Multi-Function Board (128K)	259.00
PBJ Multi-Function Board (256K)	279.00
PBJ Multi-Function Board (512K)	299.00
128K Ram Upgrade Kit (NEC)	78.00
256K Ram Upgrade (for PBJ Board)	69.00
300 Baud Modem Board	129.00
1200 Baud Modem Board	249.00
RS-232 Serial Board	89.00
Digi-Mouse/Clock Board	89.00

Tandy 1000/1200 MONITORS

Tandy VM-2 Green Monitor	129.00
Tandy CM-2 RGB Color Monitor	379.00
Tandy VM-3 TTL Green Monitor	179.00
Tatung CM-1360 Color/Gr./Amber	459.00
Tandy 1200 Text Monitor Adapt.	179.00
Tandy 1200 Color/Graphics Adapt	199.00

MODEMS

Radio Shack DCM-3 Modem	52.00
Radio Shack DCM-5 Modem	99.00
Radio Shack DC Modem 2212	315.00
Hayes Smartmodem II 300 Baud	169.00
Hayes Smartmodem 1200 Baud	429.00
Novation J-Cat 300 Baud	115.00

FOR EVERYONE

Network Four Outlet Surge Prot.	69.95
CCR-81 Cassette Recorder	52.00
CCR-82 Cassette Recorder	43.00
C-20 Digital Cassette Tapes (10pk)	9.95
Verbatim SSDD Plastic Box (10pk)	24.95
Verbatim DDDD Plastic Box (10pk)	29.95
Generic SSDD Diskettes (10pk)	17.95
NEC 64K Ram Chips (set of 8)	39.00
Radio Shack 16K Ram Chips (8)	25.00
Locking Disk Box (holds 70)	19.95
Flip N File Disk 50	15.00
Fanfold CleanEdge Paper (2600)	35.00

Radio Shack software 10% off.
Send for complete listing of
brand name software and hardware.

CALL TOLL FREE
1-800-343-8124

- LOWEST POSSIBLE PRICES
- BEST POSSIBLE WARRANTY
- KNOWLEDGEABLE SALES STAFF
- TIMELY DELIVERY
- SHOPPING CONVENIENCE



**computer
plus**
P.O. Box 1094
480 King Street
Littleton, MA 01460
SINCE 1973

IN MASSACHUSETTS CALL (617) 486-3193

*Contrasting and Comparing
BASIC, PASCAL and C*

In Defense of Other Languages

By William Barden, Jr.
PCM Contributing Editor

I had just received my copy of the December issue of PCM and was looking over the column entitled "In Defense of BASIC." Hmm . . . maybe I was a *little* strong in extolling the virtues of BASIC over other languages. The doorbell rang just then, and I stuffed the magazine under my arm and opened the door. I was greeted by two men, each with a copy of the December issue of PCM under *their* arms! One was dressed in a gray suit with blotches of chalk dust over the sides and back. The second was bearded and much more casual — he wore a tee shirt emblazoned with "Babbage Lives!", cut-off shorts and New Balance 470 running shoes.

"Are you Bill Barden?" gray suit said. I admitted to it.

"My name is Doctor Proctor, and this is Stan Smith. I teach computer science

at Twin Peaks Community College and Stan is a systems programmer at Digital Data Communicomm Systems Company. We'd like to discuss your frivolous article in this issue of PCM," he stated, waving the December issue under my nose.

"Frivolous?" I bristled.

"Well, yes. You really didn't give other languages a break. We're here to set the record straight."

Gritting my teeth, I invited both gentlemen in, and we spent the afternoon comparing notes about three languages. Dr. Proctor was an advocate of PASCAL, a language very popular in academic circles and taught at many colleges and universities. Stan was a champion of the C language, a language touted as being good for systems programming work. I, of course, was the enemy — an advocate of BASIC. The result of the discussion is presented here, an overview of PASCAL and C, contrasted with BASIC.

An Overview of Languages

As I mentioned in that article on BASIC, there have been many computer languages used for various computers over the past 30 years that digital computers have been used. Some back-

ground is helpful before getting into the mechanics of PASCAL and C.

The first language was *machine language*, coding in binary ones and zeros, or in octal or hexadecimal notation, "shorthand" versions of binary. Grace Hopper, chief developer of COBOL, maintains that nobody could code a string of more than about 30 ones and zeros without introducing an error, and this is largely true. For that reason, *assembly language* soon replaced machine language for coding. In assembly language, the binary or octal codes were replaced by more English-like *mnemonics* such as ADD for addition of two numbers and SRL for "Shift Right Logical." An assembler program acted as a translator in automatically transforming the symbolic assembly language into the equivalent ones and zeros of machine language.

However, assembly language was (and is) a very tedious language in which to code. In the late '50s, Backus of IBM developed the first "higher-level" computer language, FORTRAN. FORTRAN was very much geared to solving mathematical problems; the name itself is an acronym for "Formula Translator." A typical FORTRAN statement might look like this:

(William Barden, Jr. is a master communicator in a field in which he is one of the few recognized experts — micro-computers. A prolific author of more than 27 books on computers and computer programming, Bill also has authored several instructional software projects for Tandy/Radio Shack.)

$$A = -B + \text{SQRT}(B^2 - 4AC) / (2A)$$

FORTRAN was an instant success — it was a great deal easier to code in FORTRAN format than to write long strings of assembly language instructions. Today, FORTRAN is still used by many companies that do scientific processing.

FORTRAN was followed by a business language called COBOL (Common Business Oriented Language) in the early '60s. At the time, there was some question just how well computers could process business applications such as billing. COBOL, however, proved enormously popular, and remains so, especially in "mainframe" environments. A typical COBOL statement is very "English-like" and might appear as

MOVE NEW-PURCHASES IN THIS-MONTH TO PURCHASES.

WRITE NEW-BALANCE-1 AFTER ADVANCING TWO LINES.

A succession of languages soon appeared. PL/I (Programming Language/I) was developed by IBM and attempted to combine the best features of FORTRAN, COBOL and ALGOL, the last being

an "algorithmic language." BASIC was also developed early in the computer game by two Dartmouth college professors, Kemeny and Kurtz. There were a host of others, including LISP (List Processing), RPG (Report Program Generator) and JOVIAL (Jules' Own Version of an International Algorithmic Language).

PASCAL Rears its Head in a Cloud of Chalk Dust

PASCAL also appeared early in the history of digital computers. It was developed in the late 1960s by Professor Niklas Wirth of Zurich as a language suitable for teaching programming. PASCAL has its roots in the aforementioned ALGOL language, a language geared to cleanly implementing programming algorithms. PASCAL was, to coin a phrase, "modular and well structured," meaning that a typical program might be composed of blocks of code made up of smaller blocks of code and that each block had a well-defined, straightforward structure.

PASCAL remained relatively obscure until the late 1970s when a University of California San Diego professor, Kenneth Bowles, developed a micro-computer version of PASCAL called

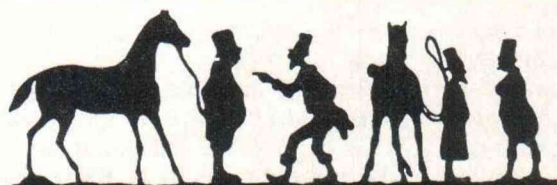
UCSD PASCAL. This implementation of PASCAL was fairly "portable" — it could be moved from one system to another by writing a short "interpreter" that translated a "pseudo-code" into the machine code used by the system on which the language was to run.

Because of UCSD PASCAL and other factors — an overall crisis involving unreadable and unmaintainable programs in industry and a true need for a good language to introduce students to computers — PASCAL became very popular in academia and remains so today.

C? Si!

Sorry, my first and last pun on C . . . C was developed in the 1970s as part of the Bell Telephone Labs Unix Operating System. As a matter of fact, almost all of Unix was *written* in the C language as are similar projects done by current systems software developers, such as Microsoft.

C has much of the same structure and appearance of PASCAL, but has more capability for "twiddling bits" — performing operations on a bit level, such as testing, masking, and shifting bits. C spans more of a range of functions than PASCAL and other higher-level lan-



The Handicapper

Use your **Tandy 1000, 1200, 2000** or **Model 100/200** to improve your performance at the track! Separate programs for **Thoroughbreds, Harness horses** and **Greyhounds** rank the horses or dogs in each race. Handicap a race in minutes, even if you've never handicapped a race before!

All the data you need is readily available in the thoroughbred *Racing Form*, harness or dog track program. We even provide diagrams showing you where to find the information you need! Data entry is quick and easy. Our manual shows you how to bet, when to bet and when to sit out—one of the real secrets of good handicapping.

Thoroughbred, Harness or Greyhound Handicapper, \$49.95 each on T1K disk. **Thoroughbred and Harness only**, \$49.95 each on **Model 100/200** tape. Any two for \$74.95, all three \$99.95. You can buy a more expensive handicapper, but you can't buy a better one!



Federal Hill Software
8134 Scotts Level Rd.
Baltimore, MD 21208



Toll Free (Orders Only) 800-245-6228
For Information Call 301-521-4886

MARYMAC INDUSTRIES INC.

800-231-3680

Radio Shack® Tandy®

EPSON PRINTERS

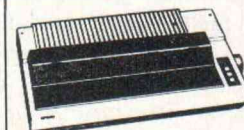
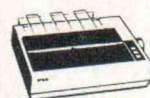
People you Trust to give you the very best!



- Lowest Discount Prices

- Reliable Service

- Quality Products



"World's largest independent authorized computer Dealer"

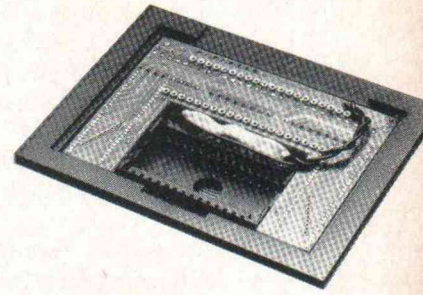
22511 Katy Fwy., Katy (Houston) Texas 77450
(713) 392-0747 Telex 774132

Give your Model 100

128K

RAM

Installs as easily
as plugging
in a socket



Software included, transfers from bank to bank. Works like main menu! Includes powerful RAM Basic that lets programs store and access data from any other bank.

PCSG says: Satisfaction guaranteed or your money back within 30 days

As amazing as it seems you can upgrade your Model 100 to 128K of RAM in just 60 seconds.

It comes to you right out of the box looking just like the picture. You just open the little compartment on the back of your Model 100 with a quarter and it just pushes right into place. You can then put the cover back in its place.

You then have 4 banks of RAM of 32K each. The additional three banks also work just like your Main Menu.

You push a function key and you are in the second bank. Push again and you are in third, again, then fourth. Press it once again for your original bank.

It has its own built-in NiCad battery that recharges right from the Model 100 and its guaranteed for a full year.

What is really great is that you can copy a file from one bank to another with just a function key.

Each bank is like having another Model 100, and all the built-in programs as well as any snap-in ROM programs appear in all four banks and work the same way. Your widebar cursor moves from file to file and you access any file or run any program just by pressing ENTER.

What lets you copy any file from one bank to another is a snap-in ROM from PCSG called RAM+, that comes at no extra charge. It just pushes right into the little socket in that same compartment with the 96K expansion unit.

Not only does this firmware let you copy a file from bank to bank, but you can make a copy of any file within the same bank instantly with a function key. Great for Lucid spreadsheets!

Copy a file from bank to bank with a function key

You can also rename a file, or kill any file with just a function key. Plus you can do a whole lot of other useful things like setting the date, day and time with function key ease. You even have a function key that lets you use non-Radio Shack printers without having to make those tricky dipswitch settings.

RAM+ lets you cold start any one of your banks without affecting the other three. That means that anytime you want you can clean out a bank's entire memory, but leave intact all the files in the other banks.

What is also fantastic is that you don't have to have the ROM in place to use the additional RAM. Whenever you take out the snap-in ROM it leaves behind a tiny machine code program that lets you switch from bank to bank just by pressing ENTER.

This lets you use your ROM socket to snap-in other ROMs like LUCID spreadsheet, WRITE ROM text processor, or DISK+ ROM file transfer program, and use them in any or all four banks. All of these, by the way, are available from PCSG.

When you are ready to copy a file from one bank to another or use any of the other fantastic functions we talked about you can just snap the RAM+ ROM back into place.

Everybody that has this 128K system in their Model 100 is so excited, because it gives them four times the capacity and all banks work just like the Main Menu.

And what has made a lot of people happy is that the system bus, located in the same compartment, is left free for you to plug in a DVI or the Holmes Engineering/PCSG portable disk drive.

The ability to copy a file from bank to bank instantly with a function key, plus all of the other features make this RAM extension truly an engineering masterpiece.

Some people hesitate when they think of installing something, and then others are skeptical that any additional hardware could be as good as the Model 100 itself. That's why we sell these 96K expansions on a 30 day trial. Simply return it within 30 days for a full refund if you are not satisfied. Priced at \$425. MC VISA COD.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

VISA / MASTER CARD / AM. EXP. / C.O.D.

guages — it offers many of the same higher-level capabilities, but also allows a programmer to get his hands dirty on a systems programming level.

Both C and PASCAL are currently among the most popular languages for microcomputers. Looking in "RSC-15," the 1986 Radio Shack Computer Center catalog, for example, you'll find a COBOL, FORTRAN, PASCAL, C and BASIC compiler, along with a Macro Assembler package. That just about says it all for current language popularity.

Let's look at the structure of PASCAL and C by comparing how things are done in those two languages versus BASIC. I'm assuming most of you are familiar enough with BASIC to follow the program logic, and you'll be able to understand the PASCAL and C versions relative to BASIC. At the end of this article, you'll have a good overview of how things work in PASCAL and C and it'll keep Proctor and Smith from appearing on my doorstep again.

Input and Output

Figure 1 shows an example of a simple input/output program written in the three languages of BASIC, PASCAL, and C. Looking at the BASIC

Figure 1. Input/Output Examples

BASIC:

```
100 INPUT "What is the first value?"; OPERAND1
110 INPUT "What is the second value?"; OPERAND2
120 TOTAL = OPERAND1 + OPERAND2
130 PRINT "The sum of "; OPERAND1; " and "; OPERAND2; is ";
    TOTAL
```

```
What is the first value? 100
What is the second value? 200
The sum of 100 and 200 is 300
```

PASCAL:

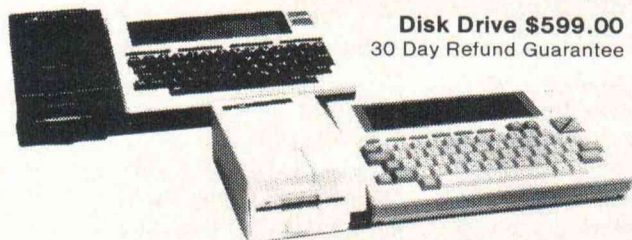
```
Var
    Operand1, Operand2, Total : integer;
begin
    write ('What is the first value? ');
    readln (Operand1);
    write ('What is the second value? ');
    readln (Operand2);
    Total := Operand1 + Operand2;
    writeln ('The sum of ', Operand1:1, ' and ', Operand2:1,
        ' is ', Total:1);
end.
```

```
What is the first value? 100
What is the second value? 200
The sum of 100 and 200 is 300
```

Personal Integrated Computers

18013 SkyPark Circle, Ste. D., Irvine, CA 92714

COMPARE PERFORMANCE



Disk Drive \$599.00
30 Day Refund Guarantee

PICDISC

- Lg. package software
- **360K** storage per disk
- **348K** file size
- **60x8** display
- Rechargeable during use & powers your computer
- OP system: **CP/M**

Direct Disk Access

TANDY

- No applic. software
- **100K** storage per disk
- **29K** max file size
- **40x8** display
- Not rechargeable
- OP system:

Load-Store Only

NOW: VIDEO INTERFACE FREE BASIC INTERPRETER

For catalog,

1-800-421-6300 literature 1-714-261-0503

Trademarks: Tandy, Tandy Corp; Cp/M-Digital Research

The Preble NavComp II

Two Aviation Programs in one package! New Enhanced Version For the Model 100 or new Tandy 200!

NavPin
For Flight Planning

- Creates Navigation Log!
- Saves Route of Flight for repeat use!
- Compensates for Climbs and Descents!
- Statute or Nautical measurements!
- Self Prompting with error prevention!
- Fly direct or use VORS!
- And many other advanced features!

NavAid
For In Flight Navigation

- Tracks your flight in Real Time!
- Modify Flight Data as necessary!
- Incredible Graphic displays!
- Shows Position, Fuel, Intersections!
- ETA'S, Ground Speed, and much more!
- Turn off and on without data loss!

Requires Radio Shack Model 100 with at least 16K or the new Tandy 200!

Attention Radio Amateurs! Announcing:

The Preble HAMTERM!

For the Model 100 or new Tandy 200!

- Radioteletype operation for BAUDOT, ASCII, AMTOR, CW!
- Split Screen Operation in receive mode with unlimited type ahead capability!
- Word Buffer in Transmit Mode! Corrects errors before they are sent!
- Automatic CQ! — Automatic Station Call! — Automatic Logging!
- Echo to printer! — Download! — Upload! — Send Time and Date! — Send RYRY!
- And many other advanced features!

Requires Kantronics UTU, RS Model 100 with at least 16K or the new Tandy 200!

The Preble NavComp: Only \$34.95

The Preble HAMTERM: Only \$29.95

Shipping and handling \$1.50 to U.S. and Canada — \$5.00 to other countries.

Free Brochures available!

VISA, MasterCard or COD is OK!

Order From: Dr. Preble's Programs; 6540 Outer Loop; Louisville, KY 40228 (502) 966-8281

version first, you can see that two values are input and assigned to variables OPERAND1 and OPERAND2. Variable TOTAL is then set equal to the sum of OPERAND1 and OPERAND2. Finally, the two operands and the result are displayed. The INPUT and PRINT keywords are used for input and output respectively. The program is in the familiar "line number" format of BASIC, with each statement preceded by a line number. Some versions of BASIC dispense with line numbers totally or partially, but we'll use the standard Microsoft MS-DOS version.

The following program performs the same functions in PASCAL. The PASCAL reserved words readln and writeln are roughly equivalent to BASIC INPUT and PRINT. The assignment of a variable to an expression is done in almost identical fashion, except that PASCAL uses a "==" (colon and equal) in place of just an "=" (equal) sign. Another thing you might notice is that PASCAL sets off statements by "begin/end" blocks and optional indents for clarity. Finally, PASCAL uses a "Var" section in which variables are defined. In this case variables Operand1, Operand2 and Total are defined as "integer" variables, meaning that they can hold whole number values from -32,768 through +32,767. PASCAL is a "strongly typed" language, meaning that all variables must be defined as certain variable types — integer, real (same as BASIC single and double-precision), character and others.

The third version of the program also produces the same result. The structure here is somewhat similar to PASCAL. The left ({} and right (}) braces define a begin/end block of code. The scanf and printf reserved words are analogous to writeln and readln of PASCAL or INPUT and PRINT of BASIC. Variable Total is set equal to the sum of Operand1 and Operand2 by an equals sign, the same as BASIC. As in PASCAL, variables Operand1, Operand2 and Total are defined as integer variable types by the "int" line. One thing that might not be obvious is that certain special characters — the "%d" characters in the scanf and printf lines — act as "place markers" in defining where the following operands are to be input and displayed.

Simple Computation

Figure 2 shows a second program written in the three languages. In this case the program performs simple computations on three numbers that are input — the sum of the three numbers

```
C:
main ()
{
  int Operand1, Operand2, Total;
  printf ("What is the first value? ");
  scanf ("%d", &Operand1);
  printf ("What is the second value? ");
  scanf ("%d", &Operand2);
  Total = Operand1 + Operand2;
  printf ("The sum of %d and %d is %d\n", Operand1, Operand2,
    Total);
}

What is the first value? 100
What is the second value? 200
The sum of 100 and 200 is 300
```

Figure 2. Computation Examples

BASIC:

```
100 PRINT "Enter three numbers separated by commas"
110 INPUT A,B,C
120 SUM = A + B + C
130 DIFF = A - B - C
140 PRODUCT = A * B * C
150 QUOTIENT = A / B / C
160 PRINT SUM, DIFF, PRODUCT, QUOTIENT

Enter three numbers separated by commas
? 12.3, 24.6, 67.8
104.7 -80.100001 20514.93 7.374631E-03
```

PASCAL:

```
var
A, B, C, Sum, Diff, Product, Quotient : real;
begin
writeln ('Enter three numbers separated by spaces. ');
readln (A, B, C);
Sum := A + B + C;
Diff := A - B - C;
Product := A * B * C;
Quotient := A / B / C;
writeln (Sum, Diff, Product, Quotient);
end.

Enter three numbers separated by spaces
12.3 24.6 67.8
1.04700000000E+02 -8.01000000000E+01 2.05149240000E+04
7.3746312684E-03
```

C:

```
main ()
{
  float A, B, C, Sum, Diff, Product, Quotient;
  printf ("Enter three numbers separated by spaces.\n");
  scanf ("%f %f %f",&A, &B, &C);
  Sum = A + B + C;
  Diff = A - B - C;
  Product = A * B * C;
  Quotient = A / B / C;
  printf ("%f %f %f %f", Sum, Diff, Product, Quotient);
}

Enter three numbers separated by spaces
12.3 24.6 67.8
104.700005 -80.100006 20514.925781 0.007375
```


$(A + B + C)$, the difference of the three numbers $(A - B - C)$, the product of the three numbers and the quotient of the three numbers $(A/B/C)$.

In the BASIC version, the logic is straightforward. First, the three numbers are read in by an INPUT statement. Variables SUM, DIFF, PRODUCT and QUOTIENT are then computed. Finally, the SUM, DIFF, PRODUCT and QUOTIENT are displayed by a PRINT.

In the PASCAL equivalent, the same sequence of steps are taken, with the code being set off by a begin/end block. Prior to the code, variables A, B, C, Sum, Diff, Product and Quotient are defined in a "Var" section to be "real" numbers. Real numbers, also known as "floating point" numbers, are the same as BASIC single- and double-precision numbers — they can be mixed numbers with integer and fractional parts, such as 34.56, 0.0056 and $-56E-5$, the latter is a form of scientific notation.

The result of the PASCAL program is expressed in floating-point format, a mixed number with an "E" and power of 10. The first result, for example, stands for 1.047 times 10 squared (1.047×100 or 104.7), the same as the BASIC result. We could have easily displayed the results in a fixed format (such as

Figure 3. Program Structure

BASIC:

```
100 '**** Program to compute area and perimeter of triangle****
110 GOTO 210
120 ' Compute Area Subroutine
130 TEMP = .5 * BASE * HEIGHT
140 PRINT "Area="; TEMP
150 RETURN
160 ' Compute Perimeter Subroutine
170 TEMP = BASE + 2 * SIDE
180 PRINT "Perimeter="; TEMP
190 RETURN
200 ' Main Program
210 PRINT "Enter Side, Base, Height"
220 INPUT SIDE, BASE, HEIGHT
230 GOSUB 130
240 GOSUB 170
250 END
```

```
Enter Side, Base, Height
? 4.5, 7.8, 16
Area= 62.4
Perimeter= 16.8
```

PASCAL:

```
(* Program to compute area and perimeter of triangle *)
```

QUAIL CANYON SYSTEMS

Serving microcomputer users since 1977

Programs for your 1000, 1200 or compatible computers

Quail Canyon Mailing List — This superbly easy to use product handles 1000 names, most any kind of labels, and will sort 1000 names in less than 30 seconds. \$34.95

Renaissance — Every game is different as you try to gain power in medieval Italy. If you like Monopoly or Risk, you'll love this. 1-4 players. \$34.95

Quail Canyon Horoscope — Calculates your complete horoscope including houses in four systems. Extremely accurate. \$29.95

Other Programs — Educational Package, Invoicing, Gradebook, Service Club Membership and Billing, Church Package, Scouting Package, Bowling League Package.

Things You Should Know — We trust our customers. Our disks are not copy protected, we don't delay orders for personal checks, we ship COD, we ship fast, our programs are all fast COM or EXE files (no slow BASIC interpreter), our software is great, our prices are low, and we'll return your money if you're not happy.

Send for our Free Catalog or order by phone, by mail, or by EMAIL on Compuserve to 70076,1463 or on Delphi to QUAIL.

**Quail Canyon Systems • (602) 885-2288
8532 E. 24th Street • Tucson, AZ 85710**

MasterCard

VISA

✓ CHECK THIS!

✓ **MY WORD!®** is a full-featured word processor with mergeprint, calculator, macros, undelete, utilities, graphics, sort; supports any printer. Printed, 180-page manual; faster than WS, with many new features. Price: just \$35!

✓ **SCC** is a spelling checker; see your errors in context and correct them. Add new words any time. Price: just \$25!

MY WORD! and **SCC** run on IBM PC, Tandy 1000/1200/2000, Sanyo 550/555 or higher with DOS 2.1, 128K, one drive.

✓ **THE CREATOR**: a database program and report generator. Sorts up to 50 keys! 200 fields, 2816 bytes/record. Select records for use with **MY WORD!** and print labels or letters. Price: only \$45!

✓ **THE CREATOR** runs on IBM PC, Tandy 1000/1200/2000 or most compatibles. (Not available for SANYO BASIC).

30-day money-back guarantee; return for refund if not satisfied.

T.N.T. SOFTWARE, INC.
34069 Hainesville Road
Round Lake IL 60073
(312) 223-8595


```

Program Triangle (Input, Output);
Var
    Side, Base, Height : Real;

Procedure ComputeA;
var
    Temp : real;
begin
    Temp := 0.5 * Base * Height;
    Writeln ('Area=', Temp);
end;

Procedure ComputeP;
var
    Temp : real;
begin
    Temp := Base + 2 * Side;
    writeln ('Perimeter=', Temp);
end;

(* main *)
begin
    writeln ('Enter Side, Base, Height');
    readln (Side, Base, Height);
    ComputeA;
    ComputeP;
end.

Enter Side, Base, Height
4.5 7.8 16
Area= 6.2400000000E+01
Perimeter= 1.6800000000E+01

```

```

C:

/* Program to compute area and perimeter of triangle */
float Side;
float Base;
float Height;

compute_area ()
{
    float Temp;
    Temp = 0.5 * Base * Height;
    printf ("Area=%f\n", Temp);
}

compute_perimeter ()
{
    float Temp;
    Temp = Base + 2 * Side;
    printf ("Perimeter=%f\n", Temp);
}

main ()
{
    printf ("Enter Side, Base, Height\n");
    scanf ("%f %f %f", &Side, &Base, &Height);
    compute_area ();
    compute_perimeter ();
}

Enter Side, Base, Height
4.5 7.8 16
Area=62.400002
Perimeter=16.800001

```

104.7) by a slightly different way of specifying the variable in the `writeln` (`Sum:1:1`).

In the C equivalent, a similar sequence of steps is followed, with `scanf` and `printf` being used in place of the PASCAL `readln` and `writeln` and BASIC's `INPUT` and `PRINT`. As in the PASCAL program, the variables `A`, `B`, `C`, `Sum`, `Diff`, `Product` and `Quotient` are first defined as floating-point (same as real variables in PASCAL or single precision in BASIC). As in PASCAL, the output is in floating-point format, which tends to spew out a lot of digits. Here again, the results could have been formatted to make their appearance closer to the BASIC version.

Note that all three languages use the same characters to represent simple arithmetic operations — a plus sign (+) for add, minus (-) for subtract, asterisk (*) for multiplication and a slash (/) for division.

Program Structure

Figure 3 shows a program that computes the area and perimeter of a triangle, given the side, base and height. It illustrates the comparative overall structure of the three languages.

The BASIC version of the program uses two subroutines, one to compute the area and one to compute the perimeter. These are called from the "main" program, which first inputs the side, base and height values and then calls the two subroutines to compute the results and print them. The first active line in the program is `GOTO 210`, which jumps over the subroutines, stored at the beginning of the code. The program could have simply used "in-line" code *without* subroutines, but this layout allows a better comparison with PASCAL and C structure.

The PASCAL version of the program produces the same results as the BASIC version, but with a somewhat different structure. Subroutines in PASCAL are called "procedures," and a large PASCAL program will be made up of many different procedures at several levels. In the PASCAL version, there is one procedure called `ComputeA` which computes the area and a second called `ComputeP` which computes the perimeter. The code within the procedures are blocked off by `begin` and `end`. The "main" code in the PASCAL program — the code which "drives" the program — makes up the third section.

Each procedure not only has code,

but may also contain its own "Var" section in which variables that are used only in the subroutine are defined. These variables are called "local" variables. One of the ground rules in PASCAL is to avoid horrendous collections of "global" variables which are changed by many different parts of the program. Having local variables helps keep the variables straight. Global variables, those which are accessible from anywhere within the PASCAL program, are defined by a beginning "Var" section.

Comments within a PASCAL program are set off by parentheses and asterisks as in (* This is a comment *). Comments can be put anywhere within the program and may extend over many lines.

The C version of the program is shown in the last part of the figure. It is structured similarly to the PASCAL version, having two "functions" that are analogous to BASIC subroutines or PASCAL procedures. The main portion of the program is the function labeled main, and the two other functions are compute area and compute perimeter. All code is blocked off by left and right brackets as before. Local variables are defined in the functions as in PASCAL. In this case we've used a floating-

point variable Temp in both functions, and the three global variables Side, Base and Height.

Comments within a C program are denoted by "/*"(slash and asterisk) as in /* This is a comment */. Like PASCAL, C comments can extend over many lines.

The most obvious differences between the structure of BASIC and the other two languages are line numbers in BASIC and their absence in PASCAL and C. Therein lies a tale. A major controversy raged for many years and is still raging about "structured programming." The BASIC GOTO statement,

Figure 4. Looping Examples

BASIC:

```
100 ***** Program to find factorials *****
110 PRINT "Enter number"
120 INPUT NUMBER
130 WNUMBER = NUMBER
140 RESULT = 1
150 WHILE (WNUMBER <> 0)
160     RESULT = RESULT * WNUMBER
170     WNUMBER = WNUMBER - 1
180 WEND
190 PRINT "Factorial of"; NUMBER; "is"; RESULT
200 RESULT = 1
210 FOR COUNTER = 1 TO NUMBER
220     RESULT = RESULT * COUNTER
230 NEXT COUNTER
240 PRINT "Factorial of"; NUMBER; "is"; RESULT
250 END
```

Enter number
? 7

DATA-BASE

Written and Distributed in BASIC Source-Code

- All progs and mods are BASIC source-code (you can modify)
- Supports fixed and variable-length records from 1 thru 252 bytes long
- Key may be anywhere in data record (key-only records ok)
- Re-uses deleted space, never needs reorganizing
- Complete documentation and a discussion of Data-Base concepts
- Can be used as a full production system
- An invaluable learning aid (includes demo progs)
- Works on all MS-DOS machines (phone for TRS-DOS info)



- | | |
|-----------------------|---|
| DB. BAS | merge into your progs to do all I/O. Supports add, replace, delete, get, get-next, find, find-next. |
| DBCRT. BAS | creates and initializes a skeleton Data-Base. |
| DBDSPLY. BAS | display/print any sector deblocked char/hex with chains. |
| DBSTATS. BAS | displays statistics plus master-index chain counts. |
| DBNDXREO. BAS | restructures and optimizes lumpy index sectors. |
| DBCMPRS. BAS | physically decreases D/B disk allocation. |
| DBALTER. BAS | alters min/max rec-len and fixed/variable characteristics. |
| DBVAL. BAS | validates all index and data keys and chains. |
| DBGGETDAT. BAS | merge into your prog to serially retrieve data records. |

- | | |
|--|------|
| <input type="checkbox"/> Data-Base System w/disk-resident documentation (requires printer) | \$49 |
| <input type="checkbox"/> Same as above plus printed documentation | \$55 |
| <input type="checkbox"/> Printed documentation only | \$8 |
| <input type="checkbox"/> Postage and diskette (not required for documentation-only) | \$5 |

checks accepted, money-orders shipped same-day

Name _____ Computer-make _____ Model _____
Address _____ Apt. _____
City _____ State _____ Zip _____

ARIES COMPUTER SOFTWARE
25357 Ingleside, Southfield, MI 48034

YOU ALREADY HAVE HALF OF THE BEST WORD PROCESSING SYSTEM IN THE WORLD

For **\$49.95** *

more, your MultiMate, Display Write, WordStar, pfs:WRITE, VolksWriter, Writing Assistant or other WP Software¹ can instantly **SET UP TO** fourteen columns of text, **create** outlines, calendars, newsletters, contracts, easy mail merges, screenplays, scripts, footnotes², **boxed copy** and lots more. It's fun with

PowerText[®]
FORMATTER

¹ Any other MS-DOS or ASCII WP Software

² On the same page or end of your document

TO ORDER, CALL OR WRITE:



BEAMAN PORTER, INC.
417 HALSTEAD AVENUE
HARRISON, NY 10528
(800) 431-0007
(914) 835-3156 IN NEW YORK

*Add \$5 for Shipping and Handling.

which, of course, transfers control anywhere within a BASIC program, is considered by many to be "harmful." We discussed this in the earlier column, pointing out examples of "spaghetti code" in BASIC which used GOTOs all over the place. A programmer can never easily follow the flow in such a program.

PASCAL and C, in their pristine states, *do not* have GOTOs. There is no way to transfer control to another part of the program directly. This *is* workable as we'll see in the next section on looping. One of the main purposes of BASIC line numbers, though, is to enable alterations of the sequence of execution of programs. Without GOTOs, line numbers are not really necessary — subroutines are *named* in PASCAL and C and invoked by name rather than line number.

Another feature that makes PASCAL and C easier to follow is the use of indentations. PASCAL and C are free-format languages — spaces are ignored and indentations can be used or not used. However, indenting for various levels and structures makes the code infinitely more understandable than BASIC listings, unless those BASIC listings also use indentations. Along with the indentations, PASCAL and C programmers are encouraged to use only one statement per physical line. This is in opposition to many BASIC programs that contain code like this:

```
100 GOSUB 10000: A(I)=A(I+1):
KEY=1: GOSUB 12000: IF LAST=A
(I+2) THEN GOSUB 12000 ELSE 1350
```

Looping

Figure 4 shows a fourth set of programs that find factorials of a number. (The factorial of the number is a simple function equal to the number times all preceding integers — the factorial of six, for example, is 1x2x3x4x5x6, or 720.)

In the BASIC version of the program, two calculations of the factorial are done, one implemented by a FOR . . . TO loop and the second by a WHILE loop.

The FOR . . . TO loop is familiar to many readers. In this case it multiplies RESULT by COUNTER and sets RESULT for COUNTER values from one through the number in question. The loop is executed six times for an input value of six, successively producing 1x1, 1x2, 2x3, 6x4, 24x5, and 120x6 for RESULT.

The WHILE . . . WEND loop may not be familiar to all readers. It is a recent addition to BASIC — an enhancement to implement some of the "structured programming" concepts we've been

```
Factorial of 6 is 720
Factorial of 6 is 720
```

PASCAL:

```
(* Program to find factorials*)
```

```
var
    Number, Wnumber, Counter, Result : Integer;
begin
    writeln ('Enter number');
    readln (Number);
    Wnumber := Number;
    Result := 1;
    while (Wnumber > 0) do
        begin
            Result := Result * Wnumber;
            Wnumber := Wnumber - 1;
        end;
    writeln ('Factorial of ', Number, ' is ', Result);
    Result := 1;
    for Counter := 1 to Number do
        begin
            Result := Result * Counter;
        end;
    writeln ('Factorial of ', Number, ' is ', Result);
end.
```

```
Enter number
7
Factorial of 7 is 5040
Factorial of 7 is 5040
```

C:

```
/* Program to find factorials*/
```

```
main ()
{
    long int Result;
    int Number, Wnumber, Counter;
    printf ("Enter number\n");
    scanf ("%d", &Number);
    Wnumber = Number;
    Result = 1;
    while (Wnumber != 0)
    {
        Result = Result * Wnumber;
        --Wnumber;
    }
    printf ("Factorial of %d is %ld\n", Number, Result);
    Result = 1;
    for ( Counter = 1; Counter <= Number; ++Counter )
    {
        Result = Result * Counter;
    }
    printf ("Factorial of %d is %ld\n", Number, Result);
}
```

```
Enter number
6
Factorial of 6 is 720
Factorial of 6 is 720
Enter number
10
Factorial of 10 is 3628800
Factorial of 10 is 3628800
```


discussing. The code between the WHILE and WEND (While END) is executed *while* a certain condition is true, in this case WNUMBER not equal to zero.

PASCAL also has a for...to loop. In the PASCAL version, the code after the for...to...do is executed in much the same manner as the BASIC loop. The while...do loop is also similar to the WHILE...WEND loop in BASIC.

In addition to these loops, PASCAL has a repeat...until loop structure, another variation on the loop theme.

Loop structures are important to PASCAL because of the lack of a GOTO. They are used in conjunction with if...then statements to provide a means to control the number of times code is repeated. Remember also that the loops may be nested to any level, just as in BASIC, providing further control.

The third version of the program shows very similar for...to and

or you could have the user data type "month:"

```
var Month : (January, February,
March, April, May, June, July,
August, September, October,
November, December); (* Pascal
*)
```

Recursive Code

Both PASCAL and C allow *recursive* code. Recursive code calls itself. A PASCAL *function*, for example, can call itself until some terminating condition is reached. This is very handy for certain types of mathematical processing. In the code in Figure 5, Power invokes itself until N = 1.

Linked Lists

PASCAL and C also have provision for manipulating linked lists and tree structures by the use of list pointers and "indirection."

Figure 5:

```
function Power(X, N : Integer) : Integer  (* Pascal *)
begin
  if N = 1
  then Power := X
  else Power := X * Power(X, N - 1)
end;
```

while loops. The differences here are that C uses the left and right braces ({ }) to mark the beginning and end of the code to be executed in the loop.

Bells and Whistles for PASCAL and C

What we've described above really only constitutes the basic components of the PASCAL and C languages. There are other features of both languages that should be considered.

Arrays

Both PASCAL and C allow numeric, character and multi-dimensional arrays, just as in BASIC. PASCAL and C are more flexible than BASIC in the ways arrays can be accessed, however. In addition, C allows arrays to be redefined in size *dynamically* (while the program is running).

User Data Types

Both PASCAL and C allow definition of user data types, a powerful feature. For example, you might define a data type "direction" to contain the values north, south, east and west:

```
enum direction north, south,
east, west } /* C */
```

- Ease of Compilation.
- Programming Efficiency.
- Speed of Execution.

Ease of Compilation

I mentioned in the previous column that BASIC might be a good choice simply because it's *there*! It's easy to load and use and comes with the system disk on Tandy MS-DOS systems. As it is an interpreter, it's also very interactive, meaning that you can easily enter, test and change BASIC code. The final debugged interpretive version can be compiled for speed.

Both PASCAL and C, however, are *compilers* (although some interpreters are available). With a compiler, you must enter the program by a separate editor, compile the program and, finally, execute the program. Discovering errors, you must re-edit the program source and go through the entire process again. It doesn't sound like too much of a chore, but in practice the extra seconds it takes to go through the loading, compilation and execution process can be a definite drag over BASIC.

One of the solutions to this is a product called *Turbo PASCAL* by Borland International. *Turbo PASCAL* (or simply "*Turbo*" as it's affectionately called) offers a very convenient environment in which to run PASCAL. With *Turbo*, you can switch instantaneously between the editor and compiler, eliminating much of the delay usually associated with a compiler. Compilation is extremely fast. The whole package is integrated, speedy and well documented. If that weren't enough, the price is only \$69.95! I've run the PC version on the Tandy 1000 with no problems and can heartily recommend it. No comparable package exists for C, to my knowledge.

Efficiency in Producing Code

Let's face it, programming is tough work! Commercial standards are 10 to 20 assembly language instructions or 60 to 100 higher level language instructions per day when all documentation, testing and debugging is considered! It becomes a necessity, therefore, to use a language in which it is easy to program. But there's another factor as well — program maintenance. Analysis of programming budgets shows that 90 percent of the money spent on in-house software is spent on program maintenance — finding and correcting program bugs after the "final" version of the program, or modifying or adding to

Bit Operators

C has bit operators — "ands," "ors," "exclusive ors," "ones complement" and "shifts" — that allow processing of fields and bits of data. This is especially handy for system programming applications.

Which Language is Best?

You can see from the foregoing that PASCAL and C offer some interesting features over BASIC. But which language is best? In all honesty, it's hard for most people to be objective about languages — it's rather like the "Fords and Chevrolets" arguments teenagers have. We'll say right at the start that no one language is *that* much better than the others. Chances are if you switch to PASCAL or C (or FORTRAN or COBOL) from BASIC, you won't find that you've discovered the ideal language for your needs. About the best you can hope for is that you've found one that you feel comfortable with and that fits your needs more precisely.

We could list in detail various factors about each of the languages, giving each a weight. Let's consider three points here that I feel really determine how useful a language is:

an existing program. There's no question in my mind that I would much rather maintain PASCAL or C than BASIC code. It's simply easier to wade through than "spaghetti."

But back to the original question. Is PASCAL or C easier to code in than BASIC? My gut feeling, backed up by test cases, is that PASCAL or C is no worse than BASIC, but not better either. I believe that BASIC code can probably be written more quickly, but that the more defined structure of PASCAL or C allows quicker and more methodical debugging.

Speed of Execution

Execution speed is another major consideration. C is reputed to be extremely fast in execution compared to other higher level languages. Certain versions of PASCAL, especially UCSD

PASCAL, proceed at a snail's pace. Turbo PASCAL claims to be very fast. It's hard to get the facts. Because execution speed is so important for a language, we ran some benchmarks with three programs using interpreter BASIC, *Microsoft Compiled BASIC*, *Turbo PASCAL* and *DeSmet C*. Although not definitive, they do point up the general range of results you can expect from language options. I think you'll be surprised at some of the results in Table 1. The three applications used for the benchmarks were:

- A 10,000-iteration loop to add, multiply and divide integer constants.
- A 1,000-iteration loop to add, multiply and divide floating-point constants.
- A worst-case bubble sort of 1,000 integers.

All three benchmarks were run on a Tandy 1000.

As you can see from the table, there are some interesting results here. Both *DeSmet C* and *Borland Turbo PASCAL* seem to be very fast in comparison to even compiled BASIC. So much so, that I will think seriously about program development in one or the other rather than BASIC. Both Proctor and Smith were overjoyed when I admitted this to them during our afternoon session, although they did get into an argument over which language, PASCAL or C, should be the one to use. For casual development and short programs, however, I'll probably stick with BASIC — the interpreter followed by the compiler, if necessary.

In spite of the execution speed differences, the comparison of the three languages points to the fact that the time is ripe for a truly innovative language, perhaps one with artificial intelligence built in, and certainly one that is more user friendly than the current ones.

I've gotta go, there's another group of irate programmers at my door, all holding copies of the December issue of PCM. See you next month, if I survive . . .

PCM

Table 1. Language Benchmarks

	Test1	Test2	Test3
Tandy 1000 BASIC:	105.0 s.	11.0 s.	4 hrs. (est.)
Microsoft BASIC Compiler:	12.0 s.	< 1.0 s.	37 mins. (est.)
Borland Turbo PASCAL:	1.6 s.	9.0 s.	134 s.
DeSmet C:	<1.0 s.	5.8 s.	137 s.

**TOLL FREE ORDER LINE
(800) 245-6228**

***** **UPGRADES** *****

**FOR MODEL 1200
and IBM PC**

MULTIFUNCTION CARD (Ser. Par. Real Time Clock Game Port 0-384K) each 64K installed	OUR "NEIGHBORHOOD PRICE COMPUTER STORE"	
	\$149.	\$759.00
	\$10.	\$95.00

Color Graphics Card (with Composite & Light Pen Ports)	\$110.	\$350.00
		NO EQUIVALENT

— FOR MODEL 1000 —

SECOND DRIVE (Exact Match)	\$124.	\$299.00
-------------------------------	---------------	-----------------

10 MEG HARD DRIVE & CONTROLLER	\$575.	\$1,000.00
--------------------------------------	---------------	-------------------

1 YR. WARRANTY
ON ALL ITEMS!!!

Call for
LATEST
PRICES!!!
(800)
245-6228



M.C. & VISA Accepted
OWL-Services
P.O. Box 116-D
Mertztown, PA.
19539
PA Res. Include 6% Tax
PA (215) 682-6855

LOW LOW PRICES
on

**Hard Disk Drives
Floppy Disk Drives
Star Printers
Anchor Modems**

**Diskettes
Color Computer Dr ϕ
SPECIAL**

IBM XT COMPATIBLE
2 FLOPPY DRIVES
1 10 MEG HARD DISK
MONOCHROME OR COLOR CARD
256 K MEMORY EXPANDIBLE TO 640 K
\$1195.00

CALL US TODAY!!! **617-278-6555** *DEALER INQUIRIES INVITED.
ORDER TOLL FREE 1-800-635-0300 **617-278-6555**



TRUE DATA PRODUCTS

9 South Main St.
Uxbridge, Mass 01569
(617) 234-7047

HOURS MON SAT 9-6 (EST)
617-278-6555

We welcome
Visa/Master Charge
Checks (allow 2 weeks for clearing)
*C.O.D. Add \$2.00

An easy way to
change printer fonts
and modes

A Handy Little Printer Setup Utility

By Marshall K. DuBois

How many times have you been at the keyboard ready to print something on your printer only to discover the printer is not set up in the special manner required? It's happened to me once too often.

In writing this helpful little utility, I determined it best to have it installed ahead of my applications program as a reminder of something out of the ordinary from the printer. Perhaps you may want to consider using this program in conjunction with a STARTER or AUTO EXEC file.

The program begins by establishing certain function labels (115-185), then resets the printer to "power-up" conditions, thus insuring all new requirements will be met. Each time the pro-

gram runs, all previous settings are lost.

After displaying the Select Menu (215-230), the program waits at the INPUT routine at Line 20. A two-phase timer/counter function allows approximately 15 seconds between selections before sounding a long tone (first phase) to remind you the program is running. This is followed by another 15 seconds before the screen clears and a message scrolls advising you the screen protection function is engaged. Touch any key and the program returns the Select Menu as you left it.

Although written for a Gemini 10, the program may be modified to accommodate most any dot matrix printer by changing the function labels (215-230) and their corresponding subroutine(s) (260-305) and (405-455) to the appropriate control codes for your printer.

Things to watch out for: In this program, selections '1' and '6' toggle (switch) between each other as do Selections '9' and '0'. Selection '7' may be used with Selection '2' or '5' only.

If you want to test your setup, do a screen print of the Select Menu after making your selections. □

(Marshall K. DuBois is President of Basic Software Inc. BSI is a software development company that specializes in the practical aspects of the computerization of small businesses. BSI also develops programs for non-scientific vertical markets.)

USE YOUR DESKTOP COMPUTER AS A DISK DRIVE FOR YOUR M-100.



Try *Disk+* for 30 days. If you aren't as excited as we are, return it for a full refund.

When we designed *Disk+* we did it out of necessity. We wanted a way that we could just connect a Model 100 to our desktop computer with a cable and save files onto the desktop's disk drive. We wanted it to be so simple to use it would be self-explanatory.

Picture this. *Disk+* comes to you on a Snap-in ROM and a diskette for your desktop. You take a quarter and open the little compartment on the back of your Model 100. Then you just press the ROM into the socket. *Disk+* appears on your main menu just like a built-in.

You connect your Model 100 to your other computer using an RS232 cable (available from PCSG for \$40).

You just place the *Disk+* diskette into the desktop's drive and turn on the computer. It powers up automatically and says "awaiting command" on your desktop's screen. Then you just put the widebar cursor on the Model 100 main menu on *Disk+* and press ENTER. You are shown your RAM files arranged just like the main menu.

To save a file to your other system's disk drive, you just move the widebar cursor to the file you want to save and press ENTER. It is saved instantly with no further action.

To look at the disk directory, you just press a function key on your Model 100. You see immediately the disk directory on your Model 100 screen, and it is arranged just like your Model 100's main menu.

To load a file from the diskette to your Model 100, you just move the widebar cursor to the file and press ENTER. The file is transferred to your Model 100's RAM instantly. You can press F8 and go back to the main menu, and the file you loaded from diskette is there, ready to use.

It is so nice to be able to keep your documents, programs (both BASIC and machine code) and *Lucid* spreadsheet files on the diskette, and bring them back when you need them. All files are ready to run or use with no changes or protocol by you.

If you have access to a desktop computer and don't have *Disk+*, then evidently we have done a poor job telling you about it.

All files and programs that you load or save, go over and come back exactly as they are supposed to be because of full error checking. This guaranteed integrity is really a comfort. *Disk+* is wonderful in so many other ways. For example, you can do a "save all" of all your RAM files with just a touch of a function key. That group of files is saved on the diskette under a single filename with a .SD (for subdirectory) extension. Any time you want, you can bring back all those files at once, or just one or two if you like, again with one-button ease.

Disk+ takes up no RAM. That's zero bytes either for storing the program or for operating overhead.

What really excites most *Disk+* users is text file cross compatibility. Your Model 100's text files are usable on your desktop computer, and your desktop's text files become Model 100 text files.

This means you can write something on your Model 100, and with *Disk+* transfer it

instantly to your desktop and start using it right away on your bigger computer. Or the way we like to work is to type in a document on the desktop computer and then transfer it to our Model 100 with *Disk+*. Then we print out the document, beautifully formatted, using WRITE ROM.

Disk+ works with just about every micro sold, from IBM PC and its clones, to all Radio Shack computers (yes, all), to Apple II, Kaypro, Epson and most CPM. Just ask us. More than likely, your computer is supported.

Incidentally, hundreds of Model 100 owners have gone to their Radio Shack stores and bought a color computer because it is so low priced, and with *Disk+* they have an inexpensive disk drive.

And if that weren't enough, how about this: *Disk+* also provides cross-compatibility between different computers like IBM, Apple or the Model 4 using the Model 100 as the intermediary device. Quite a feature!

The snap-in ROM is really great because you can use other ROMs like *Lucid* or WRITE ROM. They snap in and out as easily as an Atari game cartridge and you never lose your files in RAM.

Anyone who ever uses *Disk+* simply can't do without it. But so many times we have had new users call us and say, "Wow! I had no idea when I ordered it that *Disk+* would be so fantastic. I just couldn't believe that I could use my desktop computer's disk drive with my Model 100 just like it is another main menu."

That's why we sell *Disk+* on a thirty-day trial. If you aren't completely satisfied, return it within thirty days for a full refund. Priced at \$149.95 on Snap-in ROM. MasterCard, Visa or COD.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd., #207, Dallas, Texas 75229 • 1(214) 351-0564

PCSG provides hotline software support for the Model 100. Call us at 1-214-351-0564

© PCSG 1985

The listing:

```

0 '===== (PSU)
5 '          PRINTER SET UP UTILITY
6 '    Marshall K. DuBois - (813) 966-1252
7 '    8926 Phyliss Avenue - Sarasota, Florida 33581
8 '=====
9 ' For most any computer using MS-BASIC and dot matrix printer
10 CLS:SCREEN 0,0,0:WIDTH 80:CR$=STRING$(78,32)
15 TB$=STRING$(28,223)+"          PRINTER SET UP          "+STRING$(28,223):GOTO 100
20 '=====PRIMARY INPUT ROUTINE
25 TM=0:TC=1000
30 IN$=INKEY$: IF IN$<>" " THEN RETURN ELSE TM=TM+1:IF TM=TC THEN 35 ELSE 30
35 IF TM<2000 THEN SOUND 555,55:TC=TC+1000:GOTO 30
40 CLS:WIDTH 40:SCREEN 0
45 SR$=STRING$(33,32)+"SCREEN PROTECTION..." +STRING$(25,32)+"TOUCH ANY KEY TO RE
SUME... "
50 IN$=INKEY$:IF IN$=" " THEN GOSUB 55 ELSE CLS:WIDTH 80:PRINT TB$:GOTO 220
55 '=====SCROLL ROUTINE
60 I=1
65 LOCATE 12,1 : PRINT MID$(SR$,I,30)
70 FOR J=1 TO 30:NEXT J
75 I=I+1
80 IF I<=LEN(SR$) THEN 65
85 GOTO 50
100 '=====SELECT MENU
105 CL$=STRING$(61,32):LP=6:CLS:PRINT TB$
110 GOSUB 405:GOSUB 415:GOTO 210
115 P1$="<1> STANDARD"
120 P2$="<2> PICA          10 cpi"
125 P3$="<3> ELITE        12 cpi"
130 P4$="<4> COMPRESSED   17 cpi"
135 P5$="<6> ITALIC"
140 P6$="<5> ELONGATED     5 cpi"
145 P7$="<7> EMPHASIZED"
150 P8$="<8> DOUBLE STRIKE"
155 P9$="<9> SUPER SCRIPT"
160 P0$="<0> SUB SCRIPT"
165 L3$="<1> SET LINES PER PAGE +"
170 E1$="<E> EXIT PROGRAM"
175 L1$="<1>  6 lpi  (66 lines/pg)"
180 L2$="<2>  8 lpi  (88 lines/pg)"
185 L4$="<3>  EXIT TO PRINTER MENU"
190 RETURN
195 LOCATE 13,1:PRINT CR$:BEEP:LOCATE 13,28:SN=0:PRINT "ENTER SELECTION  -> ";:G
OSUB 20:SN=VAL(IN$):LOCATE 13,48:PRINT IN$:RETURN
200 CLS:SYSTEM
205 LOCATE 11,1:PRINT STRING$(78,223):RETURN
210 GOSUB 350
215 GOSUB 205:GOSUB 115
220 LOCATE 4,10:PRINT A$(1)P1$TAB(47)A$(7)P7$S1$;
225 LOCATE 5,10:PRINT A$(2)P2$TAB(47)A$(8)P8$S2$;:LOCATE 6,10:PRINT A$(3)P3$TAB(
47)A$(9)P9$;:LOCATE 7,10:PRINT A$(4)P4$TAB(47)A$(0)P0$;:LOCATE 8,10:PRINT A$(6)P

```



```

6$TAB(48)L3$;:LOCATE 9,10:PRINT A$(5)P5$TAB(48)E1$:LOCATE 11,1:PRINT STRING$(78,
223)
230 LOCATE 23,25 :PRINT "+ CURRENT LINES PER PAGE = "LP
235 GOSUB 195:LPRINT CHR$(7)
240 IF IN$="L" THEN SN=11 ELSE IF IN$="E" THEN SN=12 ELSE IF SN=0 THEN SN=10
245 ON SN GOTO 260,265,270,275,285,280,290,295,300,305,315,310
250 'SELECTION 1 2 3 4 5 6 7 8 9 0 L E
255 '-----SET NEW PRINTER CONDITIONS
260 LPRINT CHR$(27)CHR$(53);:GOSUB 410:GOTO 215
265 LPRINT CHR$(27)CHR$(64);:GOSUB 415:GOTO 215
270 LPRINT CHR$(27)CHR$(66)CHR$(2);:GOSUB 420:GOTO 215
275 LPRINT CHR$(27)CHR$(66)CHR$(3);:GOSUB 425:GOTO 215
280 LPRINT CHR$(27)CHR$(52);:GOSUB 430:GOTO 215
285 LPRINT CHR$(27)CHR$(87)CHR$(1);:GOSUB 435:GOTO 215
290 LPRINT CHR$(27)CHR$(64);:LPRINT CHR$(27)CHR$(69);:GOSUB 440:GOTO 215
295 LPRINT CHR$(27)CHR$(71);:GOSUB 445:GOTO 215
300 LPRINT CHR$(27)CHR$(83)CHR$(0);:GOSUB 450:GOTO 215
305 LPRINT CHR$(27)CHR$(83)CHR$(1);:GOSUB 455:GOTO 215
310 GOSUB 205:GOTO 200
315 CLS:PRINT TB$:GOSUB 205
320 LOCATE 4,26:PRINT L1$;:LOCATE 5,26:PRINT L2$;:LOCATE 6,26:PRINT L4$;:GOSUB 1
95
325 IF IN$="1" THEN LPRINT CHR$(27)CHR$(50):LP=6
330 IF IN$="2" THEN LPRINT CHR$(27)CHR$(48):LP=8
335 CLS:PRINT TB$:GOTO 220
350 '-----RESET PRINTER TO "POWER UP"
355 LPRINT CHR$(27)CHR$(18);
360 LPRINT CHR$(27)CHR$(53);
365 LPRINT CHR$(27)CHR$(87)CHR$(0);
370 LPRINT CHR$(27)CHR$(70);
375 LPRINT CHR$(27)CHR$(72);
380 LPRINT CHR$(27)CHR$(84);
385 LPRINT CHR$(27)CHR$(50);
390 LPRINT CHR$(27)CHR$(80);
395 LPRINT CHR$(27)CHR$(120)CHR$(0);:RETURN
400 '-----MOVE "*" TO SHOW CURRENT STATUS
405 A$(1)="*":A$(2)=" ":A$(3)=" ":A$(4)=" ":A$(5)=" ":A$(6)=" ":A$(7)=" ":A$(8)=
" ":A$(9)=" ":A$(0)=" ":S1$=" OFF":S2$=" OFF":RETURN
410 A$(1)="*":A$(5)=" ":RETURN
415 A$(2)="*":A$(9)=" ":A$(0)=" ":RETURN
420 A$(3)="*":A$(2)=" ":A$(4)=" ":A$(6)=" ":A$(9)=" ":A$(0)=" ":RETURN
425 A$(4)="*":A$(2)=" ":A$(3)=" ":A$(6)=" ":A$(9)=" ":A$(0)=" ":RETURN
430 A$(5)="*":A$(1)=" ":RETURN
435 A$(6)="*":A$(2)=" ":A$(3)=" ":A$(4)=" ":A$(9)=" ":A$(0)=" ":RETURN
440 A$(7)="*":A$(2)="*":A$(3)=" ":A$(4)=" ":A$(6)=" ":S1$=" ON":RETURN
445 A$(8)="*":S2$=" ON":RETURN
450 A$(9)="*":A$(2)=" ":A$(3)=" ":A$(4)=" ":A$(6)=" ":A$(0)=" ":S1$=" OFF":
S2$=" ON":RETURN
455 A$(0)="*":A$(2)=" ":A$(3)=" ":A$(4)=" ":A$(6)=" ":A$(9)=" ":S1$=" OFF":
S2$=" OFF":RETURN
460 'END OF LISTING

```



```

15 DIM A$(255),B$(255),PX(255),PY(255),SF(255),S$(25),T(255),F$(255),L(255)
20 DIM UC(255),SN$(255),RN$(255),RD$(2,255),SL(255),ST$(255),RE(255),QN(255)
22 DIM OD$(255),RX(255),RY(255),YR(255),XR(255),RT(255),RF(255),RS$(25),N$(255),
FSS(1),TP$(8),FT$(1),J(4),F2$(255),T2(255),L2(255),PN$(50),NC(50),PC(50,20),RI$(
2)
25 RESTORE 50:FOR X=1 TO 102:READ A:BIOS$=BIOS$+CHR$(A):NEXT:X!=FRE(BIOS$)
35 BU=0:BI=7:BD=0:BE=7:FI=9:FD=7:FH=10:FU=9:FE=0
40 NL$=CHR$(0):ZF$="000":SP$=SPACE$(255):FSS(0)="Reset":FSS(1)="Set"
45 FT$(0)="Off":FT$(1)="On":RESTORE 90:FOR X=1 TO 8:READ TP$(X):NEXT
50 ' Program:BIOS           Length: 102 bytes
51 DATA 85,30,7,139,236,139,94,6,255,55,157,139,94,8,139,63,139,94,10,139,55,139
,94,14,139,23,139,94,16,139,15,139,94,20,139,7,139,94,18,139,31,139,110,12,139,1
10,0,205,33,85,83,139,236,139,94,10,156,143,7,139,94,12,137,63,139,94,14
52 DATA 137,55,139,94,18,137,23,139,94,20,137,15,139,94,24,137,7,91,139,110,22,1
37,94,0,139,110,16,91,137,94,0,93,202,16,0
90 DATA Alpha/Num.,Alpha only,Time stamp,Date stamp,Integer,Single Prec.,Double
Prec.,Date entry
95 ON ERROR GOTO 400000
100 ' *** Key database information retrieval
105 OPEN"I",1,"CDBASE.PTR":INPUT #1,CD$:INPUT #1,SN:INPUT #1,CR:INPUT #1,OD:CLOS
E
110 SN$=RIGHT$(ZF$+HEX$(SN),2):CR$=RIGHT$(ZF$+HEX$(CR),2):OD$=RIGHT$(ZF$+HEX$(OD
),2):ND=CD$="00":NS=SN$="00":NP=CR$="00":NO=OD$="00"
112 OPEN"I",4,"PRINTER.PTR":MD=0
113 WHILE NOT EOF(4):MD=MD+1:INPUT #4,OD$(MD):WEND:CLOSE 4
115 IF ND THEN RUN"DBFORM"
116 IF NO THEN 120 ELSE OPEN"I",4,"DRIVER.D"+OD$:INPUT #4,CP,PM,DNS
117 FOR X=1 TO PM:INPUT #4,PN$(X),NC(X)
118 FOR XC=1 TO NC(X):INPUT #4,PC(X,XC):NEXT:CLOSE 4
120 OPEN"R",1,CD$+".MSC",2:FIELD 1,2 AS MX$
125 GET 1,1:RL=CVI(MX$):GET 1:MF=CVI(MX$):GET 1:MS=CVI(MX$):GET 1:MR=CVI(MX$)
130 GET 1:LD=ASC(LEFT$(MX$,1)):LR=ASC(RIGHT$(MX$,1))
140 OPEN"R",4,CD$+".FLD",20:FIELD 4,1 AS FL$,16 AS F$,1 AS T$,1 AS L$,1 AS UC$
145 FOR XF=1 TO MF:GET 4,XF
150 F$(XF)=LEFT$(F$,ASC(FL$)):T(XF)=ASC(T$):L(XF)=ASC(L$):UC(XF)=(ASC(UC$)>0)
155 NEXT:CLOSE 4:F=2:GOSUB 500000
156 RESTORE 157:FOR XF=MF+1 TO MF+9:READ F$(XF),T(XF),L(XF):NEXT
157 DATA *Page Number,5,4,*Record Number,5,5,*Line Number,5,5,*Delete Level,5,2,
*Tag Status,1,1,*Edit Status,1,1,*New Record Status,1,1,*Report Time,3,8,*Report
Date,1,10
160 IF LD>0 THEN OPEN"I",4,CD$+".SUB":FOR X=1 TO LD:LINE INPUT #4,SN$(X):NEXT:CL
OSE 4
165 IF LR>0 THEN OPEN"I",4,CD$+".RPT":FOR X=1 TO LR:LINE INPUT #4,RN$(X):NEXT:CL
OSE 4
200 ' *** Sub database information retrieval
205 IF NS THEN IF LD>0 THEN 400 ELSE RUN"DBFORM"
210 OPEN"R",4,CD$+".F"+SN$,4:FIELD 4,2 AS SF$,1 AS PX$,1 AS PY$
215 GET 4,1:SM=CVI(SF$)
220 FOR X=1 TO SM:GET 4:SF(X)=CVI(SF$):PX(X)=ASC(PX$):PY(X)=ASC(PY$)
225 NEXT:CLOSE 4
230 OPEN"R",4,CD$+".S"+SN$,80:FIELD 4,80 AS S$
235 FOR X=1 TO 24:GET 4,X:S$(X)=S$:NEXT:CLOSE 4
250 ' *** Report information retrieval
255 IF NP THEN 400
260 OPEN"R",4,CD$+".R"+CR$,4:FIELD 4,2 AS RF$,1 AS RX$,1 AS RY$
265 GET 4,1:RM=CVI(RF$):RW=ASC(RX$):PL=ASC(RY$)
270 GET 4:TM=ASC(LEFT$(RF$,1)):BM=ASC(RIGHT$(RF$,1)):LM=ASC(RX$):LL=ASC(RY$)
275 GET 4:BS=ASC(RX$):BB=ASC(RY$)
280 FOR X=1 TO RM:GET 4:RF(X)=CVI(RF$):RX(X)=ASC(RX$):RY(X)=ASC(RY$)

```



```

285 NEXT:CLOSE 4
290 OPEN"R",4,CD$+".P"+CR$,RW:FIELD 4,RW AS S$
295 FOR X=1 TO LL:GET 4,X:RS$(X)=S$:NEXT:CLOSE 4
300 ' *** Update disk files
305 IF ND THEN GOTO 400
310 LSET MX$=MKIS(MR):PUT 1,4:CLOSE
315 OPEN"R",1,CD$+".MSC",2:FIELD 1,2 AS MX$:F=2:GOSUB 50000
40000 ' *** Error recovery logic
40005 COLOR FD,BD:GOSUB 54500:GOSUB 52800
40010 IF ERR=71 THEN E$="Disk insertion":GOTO 40900
40020 IF ERR=72 THEN E$="Disk media":GOTO 40900
40030 IF ERR=70 THEN E$="Write protect":GOTO 40900
40040 IF ERR=61 THEN E$="Disk full":GOTO 40800
40050 IF ERR=57 THEN E$="Device I/O":GOTO 40900
40060 IF ERR=68 THEN E$="Device unavailable":GOTO 40900
40070 IF ERR=64 THEN E$="Bad file/database name":GOTO 40800
40080 IF ERR=27 THEN E$="Out of paper":GOTO 40900
40090 IF ERR=25 THEN E$="Device fault":GOTO 40900
40100 IF ERR=6 THEN E$="Data input/overflow":GOTO 40900
40110 IF ERR=7 OR ERR=14 THEN E$="Out of memory":GOTO 40800
40120 IF ERR=5 THEN E$="Illegal function call":GOTO 40800
40130 IF ERR=11 THEN E$="Division by zero error":GOTO 40800
40135 IF ERR=53 AND ERL=50910 THEN HE=-1:RESUME NEXT
40140 IF ERR=53 THEN E$="File not found":GOTO 40900
40150 IF ERR=52 THEN E$="Improper startup error":GOTO 40800
40160 E$="#"+MID$(STR$(ERR),2):GOTO 40900
40800 SL$=E$:GOSUB 54700:GOSUB 53100:RESUME 300
40900 LOCATE 25,1:PRINT E$;" error - A)bort, I)gnore, or R)etry";:SOUND 4000,1:G
OSUB 52100
40910 IF A$="A" THEN RESUME 300
40920 IF A$="I" THEN RESUME NEXT
40930 IF A$="R" THEN RESUME ELSE GOSUB 55200:GOSUB 53100:GOTO 40900
50000 '*** Open DB11 datapool file (.DAT) and record information file (.INF)
50005 XE=F/2:OPEN"R",F,CD$+".DAT",RL:X2=1:FOR X=1 TO 4:J(X)=0:NEXT
50010 FOR X=1 TO MF:ON T(X) GOTO 50015,50015,50015,50032,50020,50025,50030,50032
50015 L=L(X):GOTO 50035
50020 L=2:GOTO 50035
50025 L=4:GOTO 50035
50030 L=8:GOTO 50035
50032 L=4
50035 IF J(X2)+L>255 THEN X2=X2+1
50040 FIELD F,J(1) AS X1$,J(2) AS X2$,J(3) AS X3$,J(4) AS X4$,L AS RD$(XE,X)
50045 J(X2)=J(X2)+L:NEXT
50050 OPEN"R",F+1,CD$+".INF",1:FIELD F+1,1 AS RI$(XE):RETURN
50100 ' *** Save database record at end of file
50105 MR=MR+1:RN=MR
50200 ' *** Save database in record RN
50210 FOR XF=1 TO MF:B$(XF)=A$(XF)
50215 ON T(XF) GOTO 50220,50220,50225,50230,50235,50240,50245,50247
50220 LSET RD$(1,XF)=A$(XF):GOTO 50250
50225 LSET RD$(1,XF)=TIME$:GOTO 50250
50230 A$(XF)=DATE$:GOTO 50247
50235 LSET RD$(1,XF)=MKIS(VAL(A$(XF))):GOTO 50250
50240 LSET RD$(1,XF)=MKS$(VAL(A$(XF))):GOTO 50250
50245 LSET RD$(1,XF)=MKD$(VAL(A$(XF))):GOTO 50250
50247 LSET RD$(1,XF)=MKIS(VAL(RIGHT$(A$(XF),4)))+CHR$(VAL(LEFT$(A$(XF),2)))+CHR$
(VAL(MID$(A$(XF),4,2)))
50250 NEXT:PUT 2,RN:GOSUB 50500:RETURN
50300 ' *** Get database record RN

```



```

50305 GET 2,RN
50307 FOR XF=1 TO MF:B$(XF)=A$(XF):T=T(XF)
50310 IF T<4 THEN A$(XF)=RD$(1,XF):GOTO 50340
50312 IF T=4 OR T=8 THEN XY=CVI(LEFT$(RD$(1,XF),2)):XM=ASC(MID$(RD$(1,XF),3,1)):
XD=ASC(RIGHT$(RD$(1,XF),1)):A$(XF)=RIGHT$(ZFS+MID$(STR$(XM),2),2)+"-"+RIGHT$(ZFS
+MID$(STR$(XD),2),2)+"-"+RIGHT$(ZFS+MID$(STR$(XY),2),4):GOTO 50340
50315 IF T=5 THEN A$=RIGHT$(STR$(CVI(RD$(1,XF))),L(XF)):GOTO 50330
50320 IF T=6 THEN A$=RIGHT$(STR$(CVS(RD$(1,XF))),L(XF)):GOTO 50330
50325 IF T=7 THEN A$=RIGHT$(STR$(CVD(RD$(1,XF))),L(XF))
50330 IF LEFT$(A$,1)=" " THEN A$=MID$(A$,2)
50335 A$(XF)=A$
50340 NEXT:RETURN
50400 ' *** Get information on DB record
50405 GET 3,RN:X=ASC(RI$(1)):DR=31 AND X:NR=32 AND X:TG=64 AND X:ED=128 AND X:RE
TURN
50500 ' *** Update record RN of .INF file
50505 LSET RI$(1)=CHR$(DR OR NR OR TG OR ED):PUT 3,RN:RETURN
50600 ' *** Check if current record matches find data
50605 GET 2,RN:FOR XC=1 TO QS:ON T(QN(XC)) GOTO 50608,50608,50608,50700,50635,50
655,50675,50700
50608 ON RE(XC) GOTO 50610,50615,50620,50625,50630
50610 DF=(ST$(XC)=RD$(1,QN(XC))):GOTO 50790
50615 DF=(ST$(XC)=LEFT$(RD$(1,QN(XC)),LEN(ST$(XC)))):GOTO 50790
50620 DF=(INSTR(RD$(1,QN(XC)),ST$(XC))>0):GOTO 50790
50625 DF=(ST$(XC)>LEFT$(RD$(1,QN(XC)),LEN(ST$(XC)))):GOTO 50790
50630 DF=(ST$(XC)<LEFT$(RD$(1,QN(XC)),LEN(ST$(XC)))):GOTO 50790
50635 ON RE(XC) GOTO 50610,50610,50610,50645,50650
50645 DF=(CVI(ST$(XC))>CVI(RD$(1,QN(XC)))):GOTO 50790
50650 DF=(CVI(ST$(XC))<CVI(RD$(1,QN(XC)))):GOTO 50790
50655 ON RE(XC) GOTO 50610,50610,50610,50665,50670
50665 DF=(CVS(ST$(XC))>CVS(RD$(1,QN(XC)))):GOTO 50790
50670 DF=(CVS(ST$(XC))<CVS(RD$(1,QN(XC)))):GOTO 50790
50675 ON RE(XC) GOTO 50610,50610,50610,50685,50690
50685 DF=(CVD(ST$(XC))>CVD(RD$(1,QN(XC)))):GOTO 50790
50690 DF=(CVD(ST$(XC))<CVD(RD$(1,QN(XC)))):GOTO 50790
50700 ON RE(XC) GOTO 50610,50610,50610,50710,50720
50710 A$=RD$(1,QN(XC)):DF=(ST$(XC)>MID$(A$,2,1)+LEFT$(A$,1)+RIGHT$(A$,2)):GOTO 5
0790
50720 A$=RD$(1,QN(XC)):DF=(ST$(XC)<MID$(A$,2,1)+LEFT$(A$,1)+RIGHT$(A$,2))
50790 IF NOT DF THEN RETURN
50795 NEXT:GOTO 50307
50800 ' *** Read directory
50805 I=33:A$=LEFT$(SP$,43):GOSUB 53700:AX%=6656:GOSUB 53200
50810 AX%=19968:CX%=0:PN$=D$+"*.*.FLD"+NL$:SWAP PN$,A$:GOSUB 53700:SWAP PN$,A$:GO
SUB 53200:N=0:E=AX%:IF E>0 THEN RETURN
50815 N=N+1:X$=MID$(A$,31):N$(N)=LEFT$(X$,INSTR(X$,".")-1)
50820 AX%=20224:GOSUB 53200:E=AX%:IF E=18 THEN E=0:RETURN ELSE 50815
50900 ' *** Print help file on screen
50910 HE=0:OPEN"I",4,F$:IF HE THEN GOSUB 55900:GOSUB 53100:RETURN
50920 IF EOF(4) THEN 50950
50930 LINE INPUT #4,A$:IF LEFT$(A$,1)="\" THEN 50950
50940 PRINT A$:XL=XL+1:IF XL<22 THEN 50920
50950 LOCATE 25,13:PRINT"Hit Q or control Q to quit, any other key to continue"
50960 UC=-1:GOSUB 52000:IF A=81 OR A=17 OR EOF(4) THEN 50980
50970 CLS:XL=0:GOTO 50920
50980 CLOSE 4:CLS:RETURN
51000 ' *** Trim trailing spaces off A$
51005 L=LEN(A$):IF L=0 THEN RETURN
51010 IF MID$(A$,L,1)=" " THEN L=L-1:IF L>0 THEN GOTO 51010

```



```

51015 A$=LEFT$(A$,L):RETURN
51100 ' *** Trim trailing spaces off A$() for subdatabase
51105 FOR XS=1 TO SM:XF=SF(XS):A$=A$(XF):GOSUB 51000:A$(XF)=A$:NEXT:RETURN
52000 ' *** Input character from keyboard with UC conversion
52005 LOCATE , ,0:GOSUB 52700:IF UC AND (A>96 AND A<123) THEN A=A AND 223:A$=CH
R$(A)
52010 RETURN
52100 ' *** Input character from keyboard in UC w/printing on screen
52105 UC=-1:GOSUB 52000:IF A<32 OR A>126 THEN SOUND 4000,1:GOTO 52105
52110 PRINT A$;:RETURN
52200 ' *** Full screen advanced input routine
52205 COLOR FU,BU:IF E THEN GOSUB 51100
52207 FOR XS=1 TO SM:XF=SF(XS):IF NOT E THEN A$(XF)=""
52210 LOCATE PX(XS),PY(XS),0:PRINT A$(XF);LEFT$(SP$,L(XF)-LEN(A$(XF))):NEXT:GOSU
B 52800
52215 XS=0:IF E THEN COLOR FI,BI
52220 XS=XS+1:IF XS>SM THEN Q=0:RETURN ELSE XF=SF(XS):XP=PX(XS):YP=PY(XS):LS=LEN
(A$(XF)):XX=YP+LS:LOCATE PX(XS),XX+(LS=1 OR XX>80),0:UC=UC(XF):T=T(XF):L=L(XF)
52225 GOSUB 52000:IF A>31 AND A<127 THEN 52280
52230 IF A=4 AND (L>LS) THEN A$(XF)=A$(XF)+LEFT$(B$(XF),L-LS):LOCATE XP,YP,0:PRI
NT A$(XF);:LS=LEN(A$(XF)):XX=YP+LS:LOCATE XP,XX+(XX>80):GOTO 52225
52235 IF (A=8 OR A=29 OR A=127) AND LS>0 THEN LS=LS-1:A$(XF)=LEFT$(A$(XF),LS):LO
CATE XP,YP,0:PRINT A$(XF);LEFT$(SP$,L-LS);:LOCATE XP,LS+YP:GOTO 52225
52240 IF A=5 THEN Q=0:RETURN
52245 IF SC=71 THEN XS=0:GOTO 52220
52250 IF A=13 OR SC=80 THEN 52220
52255 IF A=17 THEN Q=-1:RETURN
52260 IF A=18 AND XS>1 AND (T=T(XS-1) OR T=1) AND L>LS THEN A$(XF)=A$(XF)+LEFT$(
A$(SF(XS-1)),L-LS):LOCATE XP,YP,0:PRINT A$(XF);:LS=LEN(A$(XF)):XX=YP+LS:LOCATE X
P,XX+(XX>80):GOTO 52225
52265 IF A=27 OR SC=135 THEN A$(XF)="":LS=0:LOCATE XP,YP,0:PRINT LEFT$(SP$,L);:L
OCATE XP,YP:GOTO 52225
52270 IF SC=72 THEN IF XS>1 THEN XS=XS-2:GOTO 52220 ELSE 52225
52275 GOTO 52225
52280 IF LS+1>L THEN SOUND 4000,1:GOTO 52225
52285 IF T=1 THEN 52305
52290 IF T>4 THEN IF A=45 OR A=46 OR (A>47 AND A<58) THEN 52305
52295 IF T=2 THEN IF A=32 OR (A>64 AND A<91) OR (A>96 AND A<123) OR A=46 THEN 52
305
52300 SOUND 4000,1:GOTO 52225
52305 A$(XF)=A$(XF)+A$:LS=LEN(A$(XF)):LOCATE XP,LS+YP-1,0:PRINT A$;:GOTO 52225
52400 ' *** Full screen editor
52405 CLS:COLOR FD,BD:FOR XR=1 TO 24:IF E=0 THEN S$(XR)=LEFT$(SP$,80) ELSE PRINT
S$(XR);
52410 NEXT:S$(25)=LEFT$(SP$,80):XR=1:XC=1:UC=0:HF=0:GOSUB 52800
52415 LOCATE XR,XC:GOSUB 52000
52420 IF A>31 AND A<127 THEN MID$(S$(XR),XC)=A$:PRINT A$;:XR=CSRLIN:XC=POS(0):GO
TO 52415
52425 IF A=4 THEN A$=MID$(S$(XR),XC+(XC>1),1):S$(XR)=LEFT$(S$(XR),XC-1)+STRING$(
81-XC,ASC(A$)):LOCATE XR,1,0:PRINT S$(XR);:GOTO 52415
52430 IF A=5 THEN FOR X=XR TO 24:S$(X)=S$(X+1):LOCATE X,1,0:PRINT S$(X);:NEXT:XR
=24:A=27
52435 IF A=8 THEN X2=(XC-2)>-1:X2=2*X2+XC:S$(XR)=LEFT$(S$(XR),X2)+RIGHT$(S$(XR),
81-XC)+"" :LOCATE XR,1,0:PRINT S$(XR);:XC=XC+(XC>1):GOTO 52415
52440 IF SC=71 THEN XR=1:XC=1:GOTO 52415
52442 IF SC=79 THEN XR=24:XC=80:GOTO 52465
52445 IF A=12 THEN FOR X=23 TO XR STEP -1:S$(X+1)=S$(X):LOCATE X+1,1,0:PRINT S$(
X+1);:NEXT:LOCATE X+1,1:A=27:GOTO 52465
52450 IF A=13 THEN XC=1:IF XR>23 THEN XR=1:GOTO 52415 ELSE XR=XR+1:GOTO 52415

```



```

52455 IF A=17 THEN RETURN
52460 IF SC=82 THEN S$(XR)=LEFT$(S$(XR),XC-1)+" "+MID$(S$(XR),XC,80-XC):LOCATE X
R,1,0:PRINT S$(XR);:GOTO 52415
52465 IF A=27 THEN S$(XR)=SPACE$(80):LOCATE XR,1,0:PRINT S$(XR);:XC=1:GOTO 52415
52470 IF SC=77 THEN XC=XC-(XC<80):GOTO 52415
52475 IF SC=75 THEN XC=XC+(XC>1):GOTO 52415
52480 IF SC=72 THEN XR=XR+(XR>1):GOTO 52415
52485 IF SC=80 THEN XR=XR-(XR<24):GOTO 52415
52490 IF SC=83 THEN S$(XR)=LEFT$(S$(XR),XC-1)+RIGHT$(S$(XR),80-XC)+" ":LOCATE XR
,1,0:PRINT S$(XR);:GOTO 52415
52495 IF SC=133 THEN XR=1:GOTO 52415
52500 IF SC=134 THEN XR=24:GOTO 52415
52505 IF SC=135 THEN XC=1:GOTO 52415
52510 IF SC=136 THEN XC=80:GOTO 52415
52515 IF SC=73 THEN HF=HF-1:IF HF<1 THEN HF=MF:GOTO 52595 ELSE 52595
52520 IF SC=81 THEN HF=HF+1:IF HF>MF THEN HF=1:GOTO 52595 ELSE 52595
52530 IF A=9 THEN IF L(HF)>80-XC THEN SOUND 4000,1:GOTO 52415 ELSE A$="":+STRING
$(L(HF),46):A=46:GOTO 52420
52590 GOTO 52415
52595 GOSUB 54500:LOCATE 25,1,0:PRINT"Field name:";F$(HF),"Length:";L(HF),"Type:
";TP$(T(HF));:GOTO 52415
52600 ' *** Input line from keyboard at current cursor position
52602 GOSUB 52800
52605 XX=CSRLIN:XY=POS(0):COLOR FU,BU:LS=LEN(I$):PRINT I$;LEFT$(SP$,L-LS);
52610 LOCATE XX,XY+LS:GOSUB 52000:IF A<32 OR A>126 THEN 52645
52615 IF LS=L THEN SOUND 4000,1:GOTO 52610
52620 IF T=1 THEN 52640
52625 IF T=2 THEN IF A=32 OR (A>64 AND A<91) OR (A>96 AND A<123) OR A=46 THEN 52
640
52630 IF T=3 THEN IF A=45 OR A=46 OR (A>47 AND A<58) THEN 52640
52635 SOUND 4000,1:GOTO 52610
52640 I$=I$+A$:LS=LS+1:PRINT A$;:GOTO 52610
52645 IF A=8 OR SC=75 OR SC=83 THEN IF LS=0 THEN SOUND 4000,1:GOTO 52610 ELSE LS
=LS-1:I$=LEFT$(I$,LS):GOTO 52660
52650 IF A=27 OR SC=135 THEN I$="":LS=0:GOTO 52660
52655 IF A=17 OR A=13 OR A=30 OR A=31 THEN COLOR FD,BD:A$=I$:GOSUB 51000:I$=A$:R
ETURN
52660 LOCATE XX,XY,0:PRINT I$;LEFT$(SP$,L-LS);:GOTO 52610
52700 ' *** Get key from keyboard queue (wait if necessary)
52705 I=22:AX%=0:GOSUB 53200:A=PEEK(VARPTR(AX%)):A$=CHR$(A):SC=PEEK(VARPTR(AX%)+
1):RETURN
52800 ' *** Flush keyboard queue
52805 I=22:AX%=768:GOTO 53200
53000 ' *** Compile formatted input screen from data in S$()
53005 SM=0:FOR XR=1 TO 24:XC=1
53010 XC=INSTR(XC,S$(XR),"."):IF XC=0 THEN 53025
53015 A=INSTR(XC,S$(XR)," "):IF A=0 THEN A=81 ELSE IF A=XC+1 THEN 53025
53020 SM=SM+1:PY(SM)=XC+1:PX(SM)=XR:SL(SM)=A-XC-1:XC=A:GOTO 53010
53025 NEXT:RETURN
53100 ' *** Pause for 2 seconds
53105 FOR P=1 TO 2500:NEXT:RETURN
53200 ' *** Call DOS function
53205 MID$(BIOS$,49,1)=CHR$(I):XV=VARPTR(BIOS$):BIOS!=PEEK(XV+1)+PEEK(XV+2)*256:
CALL BIOS!(AX%,BX%,CX%,DX%,BP%,SI%,DI%,SW%):RETURN
53300 ' *** Full screen editor
53305 CLS:COLOR FD,BD:FOR XR=1 TO 24:IF E=0 THEN RS$(XR)=LEFT$(SP$,RW) ELSE PRIN
T LEFT$(RS$(XR),80);
53310 NEXT:RS$(25)=LEFT$(SP$,RW):SP=1:XR=1:XC=1:UC=0:HF=0:GOSUB 52800
53315 LOCATE XR,XC:GOSUB 52000

```



```

53320 IF A>31 AND A<127 THEN 53450
53325 IF A=4 THEN A$=MID$(R$(X),SP+(SP>1),1):R$(X)=LEFT$(R$(X),SP-1)+STRIN
G$(RW+1-SP,ASC(A$)):LOCATE X,1,0:PRINT MID$(R$(X),SP-XC+1,80):GOTO 53315
53330 IF A=5 THEN FOR X=XR TO 24:R$(X)=R$(X+1):LOCATE X,1,0:PRINT MID$(R$(X),
SP-XC+1,80):NEXT X:XR=24:A=27
53335 IF A=8 AND SP>1 THEN SP=SP-1:XC=XC-1:R$(X)=LEFT$(R$(X),SP-1)+RIGHT$(R$
$(X),RW-SP-(SP<1))+ " ":IF XC<1 THEN 53475 ELSE LOCATE X,1,0:PRINT MID$(R$(X),
SP-XC+1,80):GOTO 53315
53340 IF SC=71 THEN XR=1:GOTO 53480
53345 IF SC=79 THEN XR=24:SP=RW:GOTO 53475
53350 IF A=12 THEN FOR X=23 TO XR STEP -1:R$(X+1)=R$(X):LOCATE X+1,1,0:PRINT M
ID$(R$(X+1),SP-XC+1,80):NEXT X:LOCATE X+1,1:A=27
53355 IF A=13 THEN XR=XR-(XR<24):GOTO 53480
53360 IF A=17 THEN RETURN
53365 IF SC=82 THEN R$(X)=LEFT$(R$(X),SP-1)+" "+MID$(R$(X),XC,RW-SP):LOCAT
E X,1,0:PRINT MID$(R$(X),SP-XC+1,80):GOTO 53315
53370 IF A=27 THEN R$(X)=LEFT$(SP$,RW):LOCATE X,1,0:PRINT LEFT$(SP$,80):GOTO
53480
53375 IF SC=77 THEN X=-(SP<RW):XC=XC+X:SP=SP+X:GOTO 53470
53380 IF SC=75 THEN X=SP>1:XC=XC+X:SP=SP+X:GOTO 53470
53385 IF SC=72 THEN XR=XR+(XR>1):GOTO 53315
53390 IF SC=80 THEN XR=XR-(XR<24):GOTO 53315
53395 IF SC=83 THEN R$(X)=LEFT$(R$(X),SP-1)+RIGHT$(R$(X),RW-SP)+" ":LOCATE
X,1,0:PRINT MID$(R$(X),SP-XC+1,80):GOTO 53315
53400 IF SC=133 THEN XR=1:GOTO 53315
53405 IF SC=134 THEN XR=24:GOTO 53315
53410 IF SC=135 THEN 53480
53415 IF SC=136 THEN SP=RW:GOTO 53475
53420 IF SC=73 THEN HF=HF-1:IF HF<1 THEN HF=MF+9:GOTO 53490 ELSE 53490
53425 IF SC=81 THEN HF=HF+1:IF HF>MF+9 THEN HF=1:GOTO 53490 ELSE 53490
53430 IF A=9 THEN IF L(HF)>RW-SP OR (T(HF)>4 AND T(HF)<8) THEN SOUND 4000,1:GOTO
53315 ELSE IF L(HF)=1 THEN A$="!":GOTO 53450 ELSE A$="\"+STRING$(L(HF)-2,45)+"\
":GOTO 53450
53435 GOTO 53315
53450 MID$(R$(X),SP)=A$:PRINT A$:IF SP>RW-1 THEN 53315 ELSE SP=SP+1:XC=XC+1
53455 IF SP/40>INT(SP/40) THEN 53315
53460 X4=INT(SP/40)*40-39:XC=40:IF X4<10 THEN 53315
53465 CLS:FOR X2=1 TO 24:LOCATE X2,1:PRINT MID$(R$(X2),X4,80):NEXT X2:GOTO 53315
53470 IF SP/40>INT(SP/40) OR XC=40 THEN 53315
53475 X4=INT(SP/40)*40-39:XC=RW-X4:IF XC>40 THEN XC=40
53477 GOTO 53465
53480 IF SP=XC THEN XC=1:SP=1:GOTO 53315 ELSE XC=1:SP=1:X4=1:GOTO 53465
53490 GOSUB 54500:LOCATE 25,1,0:PRINT"Field name: ";F$(HF),"Length: ";L(HF),"Type:
";TP$(T(HF)):GOTO 53315
53500 ' *** Compile report
53502 RT=0:FOR LL=24 TO 1 STEP -1:IF R$(LL)=LEFT$(SP$,RW) THEN NEXT:RETURN
53505 X$="#+-$*.!":FOR XR=1 TO LL:A=1
53510 IF A=RW THEN 53540 ELSE XC=A:FOR XC=XC TO RW:XT=INSTR(X$,MID$(R$(X),XC,1
)):IF XT=0 THEN NEXT XC:GOTO 53540
53515 IF XT=1 THEN A=INSTR(XC+1,R$(X),"\")+1 ELSE A=INSTR(XC,R$(X), " "):IF A
=0 THEN A=RW
53520 RT=RT+1:YR(RT)=XC:XR(RT)=XR
53525 IF XT=1 THEN MID$(R$(X),XC,A-XC)+" "+LEFT$(SP$,A-XC-2)+"\ ":RT(RT)=1:GOTO
53510
53530 IF XT=8 THEN RT(RT)=1 ELSE RT(RT)=2
53535 GOTO 53510
53540 NEXT XR:RETURN
53600 ' *** Convert Integer to signed integer
53605 IF A!>32767 THEN A%=A!-65536! ELSE A%=A!

```



```

53610 RETURN
53700 ' *** Find address for string
53705 V!=VARPTR(A$):A!=PEEK(V!+1)+PEEK(V!+2)*256:GOSUB 53600:DX%=A$:RETURN
54000 ' *** Print data in A$() on formatted screen (inverse video)
54005 COLOR FI,BI:GOTO 54110
54100 ' *** Print data in A$() on formatted screen record (normal video)
54105 COLOR FU,BU
54110 FOR XS=1 TO SM:XF=SF(XS):LOCATE PX(XS),PY(XS),0:PRINT A$(XF);LEFT$(SP$,L(XF)-LEN(A$(XF)));:NEXT:COLOR FD,BD:RETURN
54200 ' *** Print database screen
54205 COLOR FD,BD:CLS:GOSUB 54500:FOR X=1 TO 24:LOCATE X,1,0:PRINT S$(X);:NEXT:RETURN
54300 ' *** Print border line on current row of screen
54305 PRINT STRING$(80,205):RETURN
54400 ' *** Clear text on rows TW to BW
54405 AT=7:UR=TW:BR=BW:UC=1:BR=80:SL=0:GOTO 54800
54500 ' *** Clear status line
54505 COLOR FD,BD
54510 XX=CSRLIN:XY=POS(0):LOCATE 25,1,0:PRINT LEFT$(SP$,80);:LOCATE XX,XY:RETURN
54600 ' *** Print SL$ on status line (Normal)
54605 COLOR FH,BD
54610 GOSUB 54510:LOCATE 25,1:PRINT TAB((80-LEN(SL$))/2);SL$;
54615 LOCATE XX,XY:COLOR FD,BD:RETURN
54700 ' *** Print error/warning message in SL$ on status line (Inverse video)
54705 COLOR FE,BE:GOSUB 54610:SOUND 700,1:COLOR FD,BD:RETURN
54800 ' *** Scroll window up
54805 AX%=1536+SL:BX%=AT*256
54810 I=16:CX%=(UR-1)*256+(UC-1):DX%=(BR-1)*256+(BC-1):GOTO 53200
55000 SL$="--*- Field already exist --*-":GOTO 54700
55100 SL$="--*- Illegal drive --*-":GOTO 54700
55200 SL$="--*- Illegal Entry --*-":GOTO 54700
55300 SL$="--*- Too many records --*-":GOTO 54700
55600 SL$="--*- Entry too long --*-":GOTO 54700
55700 SL$="--*- Database has not been initialized - Function aborted. --*-":GOTO 54700
55800 SL$="--*- Disk drive full --*-":GOTO 54700
55900 SL$="--*- Help file not available --*-":GOTO 54700
56000 SL$="--*- Please wait --*-":GOTO 54600
56100 SL$="--*- Option has not been installed --*-":GOTO 54700
56200 SL$="--*- Y or N only --*-":GOTO 54700
56300 SL$="--*- No find parameters have been set - Function aborted --*-":GOTO 54700
56400 SL$="--*- No data in database - Function aborted --*-":GOTO 54700
56500 SL$="--*- End of file has been reached --*-":GOTO 54700
56600 SL$="--*- Beginning of file has been reached --*-":GOTO 54700
56700 SL$="--*- Function aborted --*-":GOTO 54700
56800 SL$="--*- Margin exceeds printing area --*-":GOTO 54700
56900 SL$="--*- No output driver has been defined or created --*-":GOTO 54700
57000 SL$="--*- No report was created --*-":GOTO 54700
57100 SL$="--*- Report will not fit on defined page --*-":GOTO 54700
64000 LPRINT TIMES$,DATE$:LLIST:STOP
65000 SAVE"DB11.BAS"

```

Listing 2:

```

400 ' *** Main menu for overlay
405 ' *** Current overlay = MAIN.BAS
409 ON ERROR GOTO 40000:KEY OFF:COLOR FD,BD:CLS:LOCATE 4,1,0

```



```

410 GOSUB 54300:PRINT TAB(25)"-=- DB v1.10 Main Menu -=-":GOSUB 54300:PRINT
420 PRINT TAB(29)"A)dd data to database"
430 PRINT TAB(29)"C)hange current subdatabase"
435 PRINT TAB(29)"D)ata manipulation functions"
440 PRINT TAB(29)"E)dit/View database data"
450 PRINT TAB(29)"F)ormat functions"
460 PRINT TAB(29)"P)rint/Report functions"
480 PRINT:PRINT TAB(29)"R)eturn to DOS"
490 PRINT TAB(29)"H)elp"
600 PRINT:LOCATE 18,1,0:GOSUB 54300:PRINT TAB(34)"Enter Choice:":GOSUB 52100:GO
SUB 54500
610 ON INSTR("ACDEFPRH",A$) GOTO 1000,2000,3000,4000,8000,9000,900,800
620 SOUND 4000,1:GOTO 600
800 ' *** Main Menu Help
810 F$="MAIN.HLP":GOSUB 50900:IF HE THEN 600 ELSE 400
900 ' *** Exit to DOS
910 SYSTEM
1000 '*****
1001 '*                      Add data to database                      *
1002 '*****
1003 '
1010 IF NS OR ND THEN GOSUB 55700:GOTO 600
1020 CLS:GOSUB 54200:DR=0:TG=0:ED=0:NR=32
1030 IF MR>MS THEN GOSUB 55300:GOSUB 53100:GOTO 300
1040 E=0:GOSUB 52200:IF Q THEN 300
1050 GOSUB 50100:GOTO 1030
2000 '*****
2001 '*                      Change current subdatabase                      *
2002 '*****
2003 '
2010 IF LD=0 OR ND THEN GOSUB 55700:GOTO 600
2020 CLS:GOSUB 54300:PRINT TAB(21)"-=- Change current subdatabase -=-":GOSUB
54300:PRINT
2030 PRINT"    Use the up and down arrows to scan through the available subdataba
ses."
2040 PRINT"When the desired subdatabase is found, press the space bar."
2050 PRINT"    To abort this function, press Q or control Q."
2060 X=0:COLOR FH,BD:GOTO 2160
2100 GOSUB 52000:IF A$="Q" OR A=17 THEN GOSUB 56700:GOSUB 53100:GOTO 400
2110 IF A=32 THEN 2190
2120 IF SC=72 THEN 2140
2130 IF SC=80 THEN 2160 ELSE 2100
2140 X=X-1:IF X<1 THEN X=LD
2150 GOTO 2170
2160 X=X+1:IF X>LD THEN X=1
2170 TW=11:BW=11:GOSUB 54400
2180 LOCATE 11,(80-LEN(SN$(X)))/2,0:PRINT SN$(X):GOTO 2100
2190 COLOR FD,BD:SN=X:SN$=RIGHT$(ZF$+HEX$(SN),2):NS=0
2200 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE 4
2210 GOTO 200
3000 '*****
3001 '*                      Data manipulation functions                      *
3002 '*****
3003 '
3010 GOSUB 56000:RUN"DATAMAN"
4000 '*****
4001 '*                      Edit/View database data                      *
4002 '*****

```



```

4003 '
4010 IF NS OR ND THEN GOSUB 55700:GOTO 600
4015 IF MR<1 THEN GOSUB 56400:GOTO 600
4020 AL=0:DL=16:TM=0:FM=0:JV=10:QS=0:RN=0
4030 CA$="ACDEFGHJMQSTUVWXYZ "+CHR$(17)
4035 CSC$=CHR$(71)+CHR$(79)+CHR$(72)+CHR$(80)+CHR$(75)+CHR$(77)+CHR$(133)+CHR$(1
34)+CHR$(73)+CHR$(81)+CHR$(82)+CHR$(83)
4040 CLS:GOSUB 54200:GOSUB 5100
4100 COLOR FH,BD:LOCATE 25,1,0:PRINT"Record";RN;"of";MR;" Tag:";FS$(-(TG>0));"
Edit:";FS$(-(ED>0));" New:";FS$(-(NR>0));" Modes: Tag:";FT$(-TM);" Find:";FT$
(-FM);LEFT$(SP$,80-POS(0)+1);
4190 COLOR FD,BD
4200 PR=RN:UC=-1:GOSUB 52000
4210 ON INSTR(CA$,A$) GOSUB 6100,7500,4700,6000,7000,5800,4400,4500,7300,4300,76
00,7700,6400,4600,6700,6800,6500,6600,7400,4300
4220 ON INSTR(CSC$,CHR$(SC)) GOSUB 5400,5500,5000,5100,5200,5300,5400,5500,5600,
5700,6300,6200:GOTO 4100
4300 ' *** Exit Edit/View section
4310 RETURN 300
4400 ' *** Edit/View help
4410 FS$="EDIT.HLP":GOSUB 50900:IF HE THEN RETURN 4200 ELSE GOSUB 54200:GOTO 5950
4500 ' *** Set jump value
4510 GOSUB 54500:IS$=""
4520 LOCATE 25,1,0:PRINT"Reletive jump value:";
4530 L=5:T=3:GOSUB 52600:X!=VAL(IS$)
4540 IF X!<1 OR X!>MR THEN GOSUB 55200:RETURN 4200
4550 JV=CINT(X!)-1:RETURN
4600 ' *** Set access level
4610 GOSUB 54500:IS$=""
4620 LOCATE 25,1,0:PRINT"Current access level:";AL,"New access level:";
4630 L=2:T=3:GOSUB 52600:X=VAL(IS$)
4640 IF X<0 OR X>31 THEN GOSUB 55200:RETURN 4200
4650 AL=CINT(X):RETURN
4700 ' *** Set delete level
4710 GOSUB 54500:IS$=""
4720 LOCATE 25,1,0:PRINT"Current delete level:";DL,"New delete level:";
4730 L=2:T=3:GOSUB 52600:X=VAL(IS$)
4740 IF X<0 OR X>31 THEN GOSUB 55200:RETURN 4200
4750 DL=CINT(X):RETURN
5000 ' *** View previous record that qualifies
5010 RN=RN-1:IF RN<1 THEN RN=PR:GOSUB 56600:RETURN 4200
5020 GOSUB 50400:IF AL<DR THEN 5010
5030 IF TM AND (TG=0) THEN 5010
5040 IF NOT FM THEN GOSUB 50300:GOTO 5900
5050 GOSUB 50600:IF DF THEN 5900 ELSE 5010
5100 ' *** View next record that qualifies
5110 RN=RN+1:IF RN>MR THEN RN=PR:GOSUB 56500:RETURN 4200
5120 GOSUB 50400:IF AL<DR THEN 5110
5130 IF TM AND (TG=0) THEN 5110
5140 IF NOT FM THEN 5900
5150 GOSUB 50600:IF DF THEN 5950 ELSE 5110
5200 ' *** View previous record (no matter what)
5210 RN=RN-1:IF RN<1 THEN RN=PR:GOSUB 56600:RETURN 4200
5220 GOSUB 50400:GOTO 5900
5300 ' *** View next record (no matter what)
5310 RN=RN+1:IF RN>MR THEN RN=PR:GOSUB 56500:RETURN 4200
5320 GOSUB 50400:GOTO 5900
5400 ' *** Move to beginning of file
5410 RN=0:GOTO 5100

```



```

5500 ' *** Move to end of file
5510 RN=MR+1:GOTO 5000
5600 ' *** Jump backwards JV records
5610 RN=RN-JV:GOTO 5000
5700 ' *** Jump forward JV records
5710 RN=RN+JV:GOTO 5100
5800 ' *** Goto specific record
5810 GOSUB 54500:I$=""
5820 LOCATE 25,1,0:PRINT"Record number to jump to:";
5830 L=5:T=3:GOSUB 52600:X!=VAL(I$)
5840 IF X!<1 OR X!>MR THEN GOSUB 55200:RETURN 4200
5850 RN=CINT(X!)
5900 ' *** Get record RN and print on screen
5910 GOSUB 50300
5950 ' *** Print current record on screen and return to input loop
5960 IF DR>0 THEN 54000 ELSE 54100
6000 ' *** Edit current record
6010 E=-1:GOSUB 52200:IF Q THEN GOSUB 56700:RETURN 4200 ELSE ED=128:GOTO 50200
6100 ' *** Add record to end of database
6110 IF MR>MS THEN GOSUB 55300:RETURN 4200
6120 GOSUB 54500:E=0:GOSUB 52200:IF Q THEN GOSUB 56700:RETURN 4200
6130 TG=0:DR=0:ED=0:NR=32:GOSUB 50100:RN=PR:GOTO 5900
6200 ' *** Delete record
6210 DR=DL:GOSUB 50500:GOTO 5950
6300 ' *** Unkill record
6310 DR=0:GOSUB 50500:GOTO 5950
6400 ' *** Update time/date stamp(s)
6410 GOSUB 56000:GOSUB 50200:GOTO 5900
6500 ' *** Toggle edit flag
6510 IF ED=0 THEN ED=128 ELSE ED=0
6520 GOTO 50500
6600 ' *** Toggle new record flag
6610 IF NR=0 THEN NR=32 ELSE NR=0
6620 GOTO 50500
6700 ' *** Clear all edit flags
6710 GOSUB 54500:LOCATE 25,1,0:PRINT"Are you sure you want to clear all edit flags?";
6720 GOSUB 52100:IF A$="N" THEN GOSUB 56700:RETURN 4200
6730 IF A$<>"Y" THEN GOSUB 56200:RETURN 4200 ELSE GOSUB 56000
6740 FOR RN=1 TO MR:GOSUB 50400:ED=0:GOSUB 50500:NEXT:GOTO 5400
6800 ' *** Clear all new record flags
6810 GOSUB 54500:LOCATE 25,1,0:PRINT"Are you sure you want to clear all new record flags?";
6820 GOSUB 52100:IF A$="N" THEN GOSUB 56700:RETURN 4200
6830 IF A$<>"Y" THEN GOSUB 56200:RETURN 4200 ELSE GOSUB 56000
6840 FOR RN=1 TO MR:GOSUB 50400:NR=0:GOSUB 50500:NEXT:GOTO 5400
7000 ' *** Set find parameters
7010 QS=0:I$=""
7020 QS=QS+1
7030 GOSUB 54500:LOCATE 25,1:PRINT"Field to search by (up, down, & space to choose/Q to quit):";XF=1:GOTO 7080
7040 GOSUB 52000:IF A$="Q" THEN SOUND 4000,1:RETURN
7050 IF A=32 THEN 7090
7055 IF SC=72 THEN 7065
7060 IF SC=80 THEN 7075 ELSE 7040
7065 XF=XF-1:IF XF<1 THEN XF=MF
7070 GOTO 7080
7075 XF=XF+1:IF XF>MF THEN XF=1
7080 LOCATE 25,64:PRINT F$(XF);LEFT$(SP$,80-POS(0));:GOTO 7040

```



```

7090 IS="":QN(QS)=XF:T=T(XF):IF T>4 THEN T=3 ELSE IF T>2 THEN T=1
7100 GOSUB 54500:LOCATE 25,1:PRINT"Data to search for:";
7105 L=L(XF):IF L>60 THEN L=60
7110 UC=UC(XF):GOSUB 52600:IS="":ON T(XF) GOTO 7112,7112,7112,7114,7115,7116,711
7,7114
7112 ST$(QS)=A$:GOTO 7125
7114 IF L<10 THEN GOSUB 55200:GOSUB 53100:GOTO 7100 ELSE A$=MKIS(VAL(RIGHT$(A$,
4)))+CHR$(VAL(LEFT$(A$,2)))+CHR$(VAL(MID$(A$,4,2))):ST$(QS)=MID$(A$,2,1)+LEFT$(A
$,1)+RIGHT$(A$,2):GOTO 7125
7115 ST$(QS)=MKIS(VAL(A$)):GOTO 7125
7116 ST$(QS)=MKSS(VAL(A$)):GOTO 7125
7117 ST$(QS)=MKDS(VAL(A$))
7125 GOSUB 54500:LOCATE 25,1:PRINT"Relation (1-Abs. =, 2-Left =, 3-Instring =, 4
-Less than, 5-Greater than):";
7130 L=1:T=3:GOSUB 52600:X=VAL(IS):IF X<1 OR X>5 THEN GOSUB 55200:GOSUB 53100:GO
TO 7125
7135 RE(QS)=X:IF X=1 AND T(XF)<3 THEN ST$(QS)=ST$(QS)+LEFT$(SP$,L(XF))-LEN(ST$(QS
)))
7140 GOSUB 54500:LOCATE 25,9:PRINT"Have the search paramaters for this field bee
n set correctly?";
7145 GOSUB 52100:IF A$="N" THEN 7030 ELSE IF A$<"Y" THEN GOSUB 56200:GOSUB 5310
0:GOTO 7140
7150 GOSUB 54500:LOCATE 25,17:PRINT"Are there any more paramaters to this search
?";
7155 GOSUB 52100:IF A$="Y" THEN 7020 ELSE IF A$<"N" THEN GOSUB 56200:GOSUB 5310
0:GOTO 7150
7160 RETURN
7300 ' *** Mass tag of database data that conform with find parameters
7310 IF QS<1 THEN GOSUB 56300:RETURN 4200
7320 GOSUB 54500:LOCATE 25,1,0:PRINT"Are you sure you wish to tag by the find pa
rameters?";
7330 GOSUB 52100:IF A$="N" THEN GOSUB 56700:RETURN 4200
7340 IF A$<"Y" THEN GOSUB 56200:RETURN 4200
7350 FOR RN=1 TO MR:GOSUB 50400:GOSUB 50600:IF DF THEN TG=64
7360 GOSUB 50500:NEXT:GOTO 5400
7400 ' *** Toggle tag of record
7410 IF TG=0 THEN TG=64 ELSE TG=0
7420 GOTO 50500
7500 ' *** Clear all tags in database
7510 GOSUB 54500:LOCATE 25,1,0:PRINT"Are you sure you want to clear all tags?";
7520 GOSUB 52100:IF A$="N" THEN GOSUB 56700:RETURN 4200
7530 IF A$<"Y" THEN GOSUB 56200:RETURN 4200 ELSE GOSUB 56000
7540 FOR RN=1 TO MR:GOSUB 50400:TG=0:GOSUB 50500:NEXT:GOTO 5400
7600 ' *** Toggle search view mode
7610 IF QS<1 THEN GOSUB 56300:RETURN 4200
7620 FM=(FM=0):RETURN
7700 ' *** Toggle tag view mode
7710 TM=(TM=0):RETURN
8000 '*****
8001 '* Database format functions *
8002 '*****
8003 '
8010 GOSUB 56000:RUN"DBFORM"
9000 '*****
9001 '* Print/Report functions *
9002 '*****
9003 '
9010 GOSUB 56000:RUN"PRINT"
65000 SAVE"MAIN.BAS"

```


Listing 3:

```

115 IF ND THEN 400
205 IF NS THEN 400
400 ' *** Main menu for overlay
405 ' *** Current overlay = DBFORM.BAS
406 ON ERROR GOTO 40000
409 KEY OFF:COLOR FD,BD:CLS:LOCATE 3,1,0
410 GOSUB 54300:PRINT TAB(21)"-=*=- Database format functions -=*=-":GOSUB 5430
0:PRINT
420 PRINT TAB(25)"B)uild subdatabase"
430 PRINT TAB(25)"C)hange current key database"
440 PRINT TAB(25)"D)elete database"
450 PRINT TAB(25)"F)ield name change"
460 PRINT TAB(25)"K)ill subdatabase"
470 PRINT TAB(25)"L)ist database characteristics"
480 PRINT TAB(25)"M)ake database"
490 PRINT TAB(25)"X)tend maximum record limit"
500 PRINT:PRINT TAB(25)"E)xit to DOS"
505 PRINT TAB(25)"R)eturn to main menu"
510 PRINT TAB(25)"H)elp"
600 PRINT:LOCATE 20,1,0:GOSUB 54300:PRINT TAB(34)"Enter Choice:":GOSUB 52100:GO
SUB 54500
610 ON INSTR("BCDFKLMXERH",A$) GOTO 1000,3000,5000,7000,9000,11000,13000,15000,9
50,900,800
620 SOUND 4000,1:GOTO 600
800 ' *** Main Menu Help
810 F$="DBFORM.HLP":GOSUB 50900:IF HE THEN 600 ELSE 400
900 ' *** Return to main menu
910 GOSUB 56000:RUN"MAIN"
950 ' *** Return to DOS
960 CLOSE:SYSTEM
1000 '*****
1001 '* Build subdatabase *
1002 '*****
1003 '
1010 CLS:GOSUB 54300:PRINT TAB(25)"-=*=- Build Subdatabase -=*=-":GOSUB 54300
1020 LOCATE 10,1,0:PRINT"Subdatabase name:";
1030 I$="":L=60:T=1:UC=0:GOSUB 52600:IF I$="" OR A=17 THEN 400
1040 LD=LD+1:SN$(LD)=I$
1500 E=0
1510 GOSUB 52400:CLS
1520 LOCATE 10,8,0:PRINT"Are you sure you are finished creating this subdatabase
?";
1530 GOSUB 52100:IF A$="N" THEN E=-1:GOTO 1510
1540 IF A$<>"Y" THEN GOSUB 56200:GOTO 1520
1550 CLS:LOCATE 10,21,0:PRINT"Compiling subdatabase - Please Wait...."
1560 GOSUB 53000:IF SM=0 THEN BEEP:GOTO 400
1600 XF=1:FOR XS=1 TO SM:CLS:COLOR FH,BD:PRINT S$(PX(XS));
1610 LOCATE 1,PY(XS)-1,0:COLOR FE,BE:PRINT":":COLOR FD,BD:LOCATE 2,1,0:GOSUB 54
300
1620 LOCATE 3,1,0:PRINT" Use the up and down arrows to scan through the availa
ble database fields."
1630 PRINT"When the desired field is found, press the space bar."
1640 COLOR FH,BD:GOTO 1720
1650 GOSUB 52000:IF A=32 THEN 1730
1660 IF SC=72 THEN 1680
1670 IF SC=80 THEN 1700 ELSE 1650

```



```

1680 XF=XF-1:IF XF<1 THEN XF=MF
1690 GOTO 1710
1700 XF=XF+1:IF XF>MF THEN XF=1
1710 LOCATE 11,1,0:PRINT LEFT$(SP$,80);
1720 LOCATE 11,(80-LEN(F$(XF)))/2,0:PRINT F$(XF);:GOTO 1650
1730 COLOR FD,BD:IF SL(XS)=L(XF) THEN 1770
1740 LOCATE 15,1,0:PRINT"Field lengths do not match, has the correct field been
selected?";
1750 GOSUB 52100:IF A$="N" THEN TW=15:BW=15:GOSUB 54400:GOTO 1650
1760 IF A$="Y" THEN E=-1:GOTO 1510 ELSE GOSUB 56200:GOTO 1740
1770 SF(XS)=XF:NEXT
1800 NS=0:SN=LD:SN$=RIGHT$(ZF$+HEX$(SN),2)
1810 OPEN"R",4,CD$+".F"+SN$,4:FIELD 4,2 AS SF$,1 AS PX$,1 AS PY$
1820 LSET SF$=MKI$(SM):PUT 4,1
1830 FOR XS=1 TO SM:LSET SF$=MKI$(SF(XS)):LSET PX$=CHR$(PX(XS))
1840 LSET PY$=CHR$(PY(XS)):PUT 4:NEXT:CLOSE 4
1850 OPEN"R",4,CD$+".S"+SN$,80:FIELD 4,80 AS S$
1860 FOR X=1 TO 24:LSET S$=S$(X):PUT 4,X:NEXT:CLOSE 4
1870 OPEN"O",4,CD$+".SUB":FOR X=1 TO LD:PRINT #4,SN$(X):NEXT:CLOSE 4
1880 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE 4
1890 LSET MX$=CHR$(LD)+CHR$(LR):PUT 1,5:GOTO 300
3000 *****
3001 '* Change current key database *
3002 *****
3003 '
3010 CLS:GOSUB 54300:PRINT TAB(19)"-=-*- Change current key database -=-*-":GOSU
B 54300:IS=""
3020 LOCATE 11,:PRINT"Which drive contains the new database to select?";
3030 L=1:T=2:UC=-1:GOSUB 52600:IF IS="" OR A=17 THEN 400
3035 IF IS<"A" OR IS>"Z" THEN GOSUB 55100:GOTO 3020
3040 GOSUB 54500:TW=11:BW=11:GOSUB 54400:D$=IS:GOSUB 50800
3050 IF E=0 THEN 3100
3060 IF E=18 THEN LOCATE 11,24:PRINT"No databases found on this drive":GOTO 3080
3070 LOCATE 11,24:PRINT"Error in reading disk directory!"
3080 LOCATE 13,23:PRINT"Hit any key to return to the menu":GOSUB 52000:GOTO 400
3100 XD=1
3120 LOCATE 5,1,0:PRINT" Use the up and down arrows to scan through the availa
ble databases, Q or control Q to quit."
3130 PRINT"When the desired database is found, press the space bar."
3140 COLOR FH,BD:GOTO 3220
3150 GOSUB 52000:IF A=32 THEN 3230
3155 IF A=17 OR A=81 THEN 400
3160 IF SC=72 THEN 3180
3170 IF SC=80 THEN 3200 ELSE 3150
3180 XD=XD-1:IF XD<1 THEN XD=N
3190 GOTO 3210
3200 XD=XD+1:IF XD>N THEN XD=1
3210 LOCATE 11,1,0:PRINT LEFT$(SP$,80);
3220 LOCATE 11,(80-LEN(N$(XD)))/2,0:PRINT N$(XD);:GOTO 3150
3230 CD$=D$+":"+N$(XD):SN=0:CR=0:SM=0
3240 TW=4:BW=11:GOSUB 54400
3250 LOCATE 11,22:PRINT"Loading database....Please wait...."
3260 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE:RUN 10
5000 *****
5001 '* Delete database *
5002 *****
5003 '

```



```

5010 IF ND THEN GOSUB 55700:GOTO 600
5015 CLS:GOSUB 54300:PRINT TAB(22)"-=-*- Delete current database -=-*-":GOSUB 54
300:SOUND 4000,1
5020 GOSUB 52800:LOCATE 11,13:PRINT"Are you sure you wish to delete the current
database?";
5030 L=1:T=2:UC=-1:GOSUB 52600:IF I$="N" OR A=17 THEN 400
5040 IF I$<>"Y" THEN GOSUB 56200:GOTO 5020
5050 GOSUB 54500:TW=11:BW=11:GOSUB 54400:LOCATE 11,22:PRINT"Deleting database...
Please wait...."
5060 CLOSE:KILL CD$+".*":NS=-1:ND=-1:NP=-1
5070 OPEN"O",1,"CDBASE.PTR":FOR X=1 TO 4:PRINT #1,"00":NEXT:CLOSE:GOTO 400
7000 *****
7001 '* Field name change *
7002 *****
7003 '
7010 IF ND THEN GOSUB 55700:GOTO 600
7015 CLS:GOSUB 54300:PRINT TAB(25)"-=-*- Change field name -=-*-":GOSUB 54300
7020 LOCATE 5,1,0:PRINT" Use the up and down arrows to scan through the availa
ble database fields, Q or control Q to quit."
7030 PRINT"When the desired field is found, press the space bar."
7040 XF=1:COLOR FH,BD:GOTO 7110
7050 GOSUB 52000:IF A=32 THEN 7130
7055 IF A=17 OR A=81 THEN 400
7060 IF SC=72 THEN 7080
7070 IF SC=80 THEN 7100 ELSE 7050
7080 XF=XF-1:IF XF<1 THEN XF=MF
7090 GOTO 7110
7100 XF=XF+1:IF XF>MF THEN XF=1
7110 LOCATE 11,1,0:PRINT LEFT$(SP$,80);
7120 LOCATE 11,(80-LEN(F$(XF)))/2,0:PRINT F$(XF);:GOTO 7050
7130 LOCATE 14,1:PRINT"What is the new name for this field:";
7140 L=16:T=1:UC=0:GOSUB 52600:IF I$="" THEN GOSUB 55200:GOTO 7130
7150 FOR X=1 TO MF:IF X<>XF THEN IF I$=F$(X) THEN GOSUB 55000:GOTO 7130
7160 NEXT:F$(XF)=I$:OPEN"R",4,CD$+".FLD",20:FIELD 4,1 AS FL$,16 AS F$,1 AS T$,1
AS L$,1 AS UC$
7170 LSET FL$=CHR$(LEN(I$)):LSET F$=I$:LSET T$=CHR$(T(XF)):LSET L$=CHR$(L(XF)):L
SET UC$=CHR$(-UC(XF))
7180 PUT 4,XF:CLOSE 4:GOTO 400
9000 *****
9001 '* Kill subdatabase *
9002 *****
9003 '
9010 IF NS OR ND THEN GOSUB 55700:GOTO 600
9020 CLS:GOSUB 54300:PRINT TAB(26)"-=-*- Kill Subdatabase -=-*-":GOSUB 54300:PRI
NT
9030 PRINT" Use the up and down arrows to scan through the available subdataba
ses."
9040 PRINT"When the desired subdatabase is found, press the space bar."
9050 PRINT" To abort this function, press Q or control Q."
9060 X=0:COLOR FH,BD:GOTO 9160
9100 GOSUB 52000:IF A$="Q" OR A=17 THEN 400
9110 IF A=32 THEN 9190
9120 IF SC=72 THEN 9140
9130 IF SC=80 THEN 9160 ELSE 9100
9140 X=X-1:IF X<1 THEN X=LD
9150 GOTO 9170
9160 X=X+1:IF X>LD THEN X=1
9170 TW=11:BW=11:GOSUB 54400
9180 LOCATE 11,(80-LEN(SN$(X)))/2,0:PRINT SN$(X);:GOTO 9100

```



```

9190 LOCATE 14,15,0:PRINT"Are you sure you want to delete this subdatabase?";
9200 GOSUB 52100:IF A$="N" THEN TW=14:BW=14:GOSUB 54400:GOTO 9100
9210 IF A$<"Y" THEN GOSUB 56200:GOTO 9190
9220 X$=RIGHT$(ZF$+HEX$(X),2)
9230 KILL CD$+".F"+X$:KILL CD$+".S"+X$:LD=LD-1:IF X>LD THEN 9300
9240 FOR XS=X TO LD:X$=RIGHT$(ZF$+HEX$(XS),2):X2$=RIGHT$(ZF$+HEX$(XS+1),2)
9250 NAME CD$+".F"+X2$ AS CD$+".F"+X$
9260 NAME CD$+".S"+X2$ AS CD$+".S"+X$:SN$(XS)=SN$(XS+1):NEXT
9270 X$=RIGHT$(ZF$+HEX$(LD+1),2):KILL CD$+".F"+X$:KILL CD$+".S"+X$
9300 LSET MX$=CHR$(LD)+CHR$(LR):PUT 1,5
9310 OPEN"O",4,CD$+".SUB":FOR X2=1 TO LD:PRINT #4,SN$(X2):NEXT:CLOSE 4
9320 IF SN>X THEN SN=SN-1:GOTO 9350
9330 IF SN=X THEN SN=0:NS=-1:GOTO 9350
9340 GOTO 300
9350 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLOSE #4
9450 IF NOT NS THEN 300
9460 TW=4:BW=14:GOSUB 54400
9470 LOCATE 11,2:PRINT"Since you have deleted the current subdatabase, make sure
to select a new one!"
9480 LOCATE 13,25:PRINT"Hit any key to return to menu":GOSUB 52000:GOTO 300
11000 *****
11001 * List database characteristics *
11002 *****
11003 '
11010 IF ND THEN GOSUB 55700:GOTO 600
11015 CLS:GOSUB 54300:PRINT TAB(20)"-*- List database characteristics -*-":GOSUB 54300
11020 LOCATE 11,18:PRINT"Do you want the output to go to the printer?":GOSUB 52100:GOSUB 54500
11030 IF A=13 OR A=17 OR A$="Q" THEN 400
11040 IF A$="Y" THEN OD$="LPT1":EL=52:UM=3:LOCATE 13,20:PRINT"Hit any key when
the printer is ready":GOSUB 52000:TW=11:BW=13:GOSUB 54400:LOCATE 11,28,0:PRINT"
-=- Printing -=-":GOTO 11100
11050 IF A$="N" THEN OD$="SCRN":EL=18:UM=0:CLS:GOTO 11100
11060 GOSUB 56200:GOTO 11020
11100 OPEN OD$ FOR OUTPUT AS 4:LN=0
11200 FOR XF=1 TO MF:IF LN=0 THEN PRINT #4,STRING$(UM,10):GOSUB 12000:PRINT #4,"
Fieldname";TAB(28);"Length","Type","LC allowed":GOSUB 12000:LN=3
11210 PRINT #4,F$(XF);TAB(28);L(XF),TP$(T(XF)),
11220 IF T(XF)>2 THEN PRINT #4, ELSE IF UC(XF) THEN PRINT #4,"No" ELSE PRINT #4,
"Yes"
11230 LN=LN+1:IF LN<EL THEN 11300
11240 GOSUB 12000:LN=0
11250 IF UM=0 THEN PRINT #4,TAB(28)"Hit any key to continue":GOSUB 52000:CLS:GO
TO 11300
11260 PRINT #4,CHR$(12);
11300 NEXT:GOSUB 12000:PRINT #4,"Record Length:";RL,"Number of fields:";MF
11310 PRINT #4,"Number of records:";MR,"Datapool size:";MR*RL
11320 IF UM=0 THEN PRINT #4,TAB(25)"Hit any key to return to menu":GOSUB 52000
11330 CLOSE 4:GOTO 400
12000 PRINT #4,STRING$(79,"-"):RETURN
13000 *****
13001 * Make database *
13002 *****
13003 '
13010 CLS:GOSUB 54300:PRINT TAB(27)"-*- Make database -*-":GOSUB 54300
13020 LOCATE 10,27,0:PRINT"Database name:";
13030 L=8:T=1:UC=-1:GOSUB 52600:IF I$="" OR A=17 THEN 400

```



```

13040 CD$=I$:I$="":IF NOT ND THEN CLOSE
13050 LOCATE 12,22,0:PRINT"Drive for database:";
13060 L=1:T=2:GOSUB 52600:IF I$="" THEN GOSUB 55200:GOTO 13050
13070 CD$=I$+"":CD$:GOSUB 54500:I$="":TW=10:BW=12:GOSUB 54400
13080 LOCATE 11,18,0:PRINT"Maximum number of records allowed in database:";
13090 L=5:T=3:GOSUB 52600:X!=VAL(I$)
13100 IF X!>32767 THEN GOSUB 55300:GOTO 13080
13110 IF X!<1 THEN GOSUB 55200:GOTO 13080
13120 MS=CINT(X!):MF=0:RL=0:MR=0:LD=0:LR=0
13200 GOSUB 54500:I$="":MF=MF+1:TW=4:BW=24:GOSUB 54400
13210 LOCATE 4,1,0:PRINT"Enter the following information for each field:"
13220 LOCATE 6,1,0:PRINT"Field name:";
13230 L=15:T=1:UC=0:GOSUB 52600:IF MF<2 THEN 13260
13240 FOR X=1 TO MF-1:IF I$=F$(X) THEN GOSUB 55000:GOTO 13220
13250 NEXT
13260 F$(MF)=I$:GOSUB 54500:I$=""
13270 LOCATE 8,1,0:PRINT"Valid field data types:"
13280 PRINT,"1) Alpha/numeric":PRINT,"2) Alpha only"
13290 PRINT,"3) Automatic time stamp":PRINT,"4) Automatic date stamp"
13300 PRINT,"5) Integer number (-32767>#>32767)"
13310 PRINT,"6) Single precision number (7 significant digits)"
13320 PRINT,"7) Double precision number (16 significant digits)"
13325 PRINT,"8) Date entry (MM-DD-YYYY)"
13330 LOCATE 18,1,0:PRINT"Enter the number for the data type for this field:";
13340 L=1:T=3:GOSUB 52600:X=VAL(I$)
13350 IF X<1 OR X>8 THEN GOSUB 55200:GOTO 13270
13360 I$="":T(MF)=X:GOSUB 54500
13370 IF X=3 THEN L(MF)=8:UC(MF)=0:GOTO 13482
13380 IF X=4 OR X=8 THEN L(MF)=10:UC(MF)=0:GOTO 13482
13390 IF X>4 THEN UC(MF)=0:GOTO 13450
13400 LOCATE 19,1,0:PRINT"Should lower case letters be allowed to be entered?";
13410 L=1:T=2:UC=-1:GOSUB 52600
13420 IF I$="Y" THEN UC(MF)=0:GOTO 13440
13430 IF I$="N" THEN UC(MF)=-1 ELSE GOSUB 56200:GOTO 13400
13440 GOSUB 54500:I$=""
13450 LOCATE 20,1,0:PRINT"Maximum input length (79 max.):";
13460 L=2:T=3:GOSUB 52600:X=VAL(I$)
13470 IF X<1 OR X>79 THEN GOSUB 55200:GOTO 13450
13480 L(MF)=X:GOSUB 54500
13482 ON T(MF) GOTO 13484,13484,13484,13486,13485,13486,13487,13486
13484 RL=RL+L(MF):GOTO 13490
13485 RL=RL+2:GOTO 13490
13486 RL=RL+4:GOTO 13490
13487 RL=RL+8
13490 LOCATE 22,23,0:PRINT"Is the above information correct?";
13500 GOSUB 52100
13510 IF A$="N" THEN MF=MF-1:GOTO 13200
13520 IF A$<>"Y" THEN GOSUB 56200 ELSE GOSUB 54500
13530 LOCATE 23,18,0:PRINT"Are there any more fields in this database?";
13540 GOSUB 52100
13550 IF A$="Y" THEN 13200
13560 IF A$<>"N" THEN GOSUB 56200
13570 TW=4:BW=25:GOSUB 54400:LOCATE 12,21,0:PRINT"Initializing database - Please
wait...."
13600 OPEN"R",1,CD$+".MSC",2:FIELD 1,2 AS MX$
13610 LSET MX$=MKI$(RL):PUT 1,1:LSET MX$=MKI$(MF):PUT 1
13620 LSET MX$=MKI$(MS):PUT 1:LSET MX$=MKI$(MR):PUT 1
13630 LSET MX$=CHR$(LD)+CHR$(LR):PUT 1
13640 ND=0:NS=-1:NP=-1:SN=0:CR=0

```



```

13650 OPEN"R",2,CD$+" .FLD",20:FIELD 2,1 AS FL$,16 AS F$,1 AS T$,1 AS L$,1 AS UC$
13660 FOR XF=1 TO MF:LSET FL$=CHR$(LEN(F$(XF))):LSET F$=F$(XF)
13670 LSET T$=CHR$(T(XF)):LSET L$=CHR$(L(XF)):LSET UC$=CHR$(-UC(XF))
13680 PUT 2,XF:NEXT:CLOSE 2
13690 OPEN"O",2,"CDBASE.PTR":PRINT #2,CD$:PRINT #2,SN:PRINT #2,CR:PRINT #2,OD:CL
OSE 2
13700 GOTO 300
15000 *****
15001 '*                      Extend maximum record limit                      *
15002 *****
15003 '
15010 IF NS OR ND THEN GOSUB 55700:GOTO 600
15020 CLS:GOSUB 54300:PRINT TAB(25)"-=-*- Extend maximum record limit -=-*-":GOS
UB 54300
15030 LOCATE 5,1:PRINT"Current maximum record limit is set at";MS:I$=""
15040 LOCATE 11,1:PRINT"What is the new maximum record limit?";
15050 L=5:T=3:GOSUB 52600:IF I$="" OR A=17 THEN 400 ELSE X!=VAL(I$)
15060 IF X!>32767 THEN GOSUB 55300:GOTO 15040
15070 IF X!<0 THEN GOSUB 55200:GOTO 15040
15080 MS=CINT(X!):LSET MX$=MKI$(MS):PUT 1,3:GOTO 300
65000 SAVE "DBFORM.BAS"

```

Listing 4:

```

400 ' *** Main menu for overlay
405 ' *** Current overlay = DATAMAN.BAS
409 KEY OFF:COLOR FD,BD:CLS:LOCATE 4,1,0
410 GOSUB 54300:PRINT TAB(20)"-=-*- Data manipulation functions -=-*-":GOSUB 54
300:PRINT
420 PRINT TAB(29)"E)rase database data"
430 PRINT TAB(29)"O)rganize database"
440 PRINT TAB(29)"S)ort database"
450 PRINT TAB(29)"T)ransfer database data"
500 PRINT:PRINT TAB(29)"R)eturn to main menu"
510 PRINT TAB(29)"H)elp"
600 PRINT:LOCATE 16,1,0:GOSUB 54300:PRINT TAB(34)"Enter Choice: ";:GOSUB 52100:GO
SUB 54500
610 ON INSTR("EOSTRH",A$) GOTO 1000,3000,5000,7000,900,800
620 SOUND 4000,1:GOTO 600
800 ' *** Main Menu Help
810 F$="DATAMAN.HLP":GOSUB 50900:IF HE THEN 600 ELSE 400
900 ' *** Return to main menu
910 GOSUB 56000:RUN"MAIN"
1000 *****
1001 '*                      Erase database data                      *
1002 *****
1003 '
1010 IF ND THEN GOSUB 55700:GOTO 600
1015 CLS:GOSUB 54300:PRINT TAB(20)"-=-*- Erase current database data -=-*-":GOSU
B 54300:SOUND 4000,1
1020 GOSUB 52800:LOCATE 11,6:PRINT"Are you sure you wish to delete the data in t
he current database?";
1030 L=1:T=2:UC=-1:GOSUB 52600:IF I$="N" OR A=17 THEN 400
1040 IF I$<>"Y" THEN GOSUB 56200:GOTO 1020
1050 GOSUB 54500:TW=11:BW=11:GOSUB 54400:LOCATE 11,22:PRINT"Deleting data...Plea
se wait...."
1060 CLOSE 2,3:KILL CD$+" .DAT":MR=0:GOTO 300
3000 *****

```



```

3001 '*                      Organize database                      *
3002 '*****
3003 '
3010 IF ND THEN GOSUB 55700:GOTO 600 ELSE IF MR=0 THEN GOSUB 56400:GOTO 600
3015 CLS:GOSUB 54300:PRINT TAB(20)"-==*- Organize database -==*-":GOSUB 54300:IS
=="
3020 LOCATE 11,1:PRINT"What is the highest delete level to be immune from elimin
ation?";
3030 L=2:T=3:GOSUB 52600:IF A=17 OR IS="" THEN 400
3040 AL=VAL(IS):IF AL<1 OR AL>31 THEN GOSUB 55200:GOTO 3020
3050 GOSUB 54500:TW=11:BW=11:GOSUB 54400
3060 LOCATE 11,21:PRINT"Organizing database....Please wait...."
3070 CR=0
3100 FOR RN=1 TO MR:GOSUB 50400:IF AL<DR THEN 3130
3110 CR=CR+1:IF CR=RN THEN 3130
3120 GET 2,RN:PUT 2,CR:GET 3,RN:PUT 3,CR
3130 NEXT:MR=CR:GOTO 300
5000 '*****
5001 '*                      Sort database                      *
5002 '*****
5003 '
5010 IF ND THEN GOSUB 55700:GOTO 600 ELSE IF MR=0 THEN GOSUB 56400:GOTO 600
5015 CLS:GOSUB 54300:PRINT TAB(28)"-==*- Sort database -==*-":GOSUB 54300
5020 LOCATE 5,1:PRINT" Use the up and down arrows to scan through the available
database fields,":PRINT"Q or Control Q to quit."
5030 PRINT" When the desired field for the current sort priority is found, pres
s the":PRINT"space bar. Remember, low sort priorities are the most significant.
"
5040 LOCATE 11,8:PRINT"Current field:":LOCATE 13,1:PRINT"Current sort priority:"
:X=1:X2=0:COLOR FH,BD:LOCATE 13,36:PRINT X2+1;"of";MF
5045 FOR XX=1 TO MF:X(X)=0:NEXT:GOTO 5130
5050 GOSUB 52000:IF AS="Q" OR A=17 THEN 400
5060 IF A=32 THEN 5140
5070 IF SC=72 THEN 5090
5080 IF SC=80 THEN 5110 ELSE 5050
5090 X=X-1:IF X<1 THEN X=MF
5100 IF X(X) THEN 5090 ELSE 5130
5110 X=X+1:IF X>MF THEN X=1
5120 IF X(X) THEN 5110
5130 LOCATE 11,35:PRINT F$(X);LEFT$(SP$,16);:GOTO 5050
5140 X2=X2+1:SP(X2)=X:X(X)=-1:IF X2<MF THEN LOCATE 13,36:PRINT X2+1;"of";MF:GOTO
5110
5150 TW=5:BW=13:GOSUB 54400:LOCATE 12,22:PRINT"Sorting database....Please wait..
..";
5160 F=4:GOSUB 50000:M=MR
5200 M=INT(M/2):IF M=0 THEN 300
5210 J=1:K=MR-M
5220 I=J
5230 L=I+M
5240 GET 2,I:GET 4,L:FOR X=1 TO MF:X2=SP(X)
5250 IF RD$(1,X2)=RD$(2,X2) THEN 5290
5260 ON T(X2) GOTO 5265,5265,5265,5270,5275,5280,5285,5270
5265 IF RD$(1,X2)<RD$(2,X2) THEN 5320 ELSE 5300
5270 IF (MID$(RD$(1,X2),2,1)+LEFT$(RD$(1,X2),1)+RIGHT$(RD$(1,X2),2))<(MID$(RD$(2
,X2),2,1)+LEFT$(RD$(2,X2),1)+RIGHT$(RD$(2,X2),2)) THEN 5320 ELSE 5300
5275 IF CVI(RD$(1,X2))<CVI(RD$(2,X2)) THEN 5320 ELSE 5300
5280 IF CVS(RD$(1,X2))<CVS(RD$(2,X2)) THEN 5320 ELSE 5300
5285 IF CVD(RD$(1,X2))<CVD(RD$(2,X2)) THEN 5320 ELSE 5300
5290 NEXT:GOTO 5320

```



```

5300 GET 3,I:GET 5,L:PUT 2,L:PUT 3,L:PUT 4,I:PUT 5,I
5310 I=I-M:IF I>0 THEN 5230
5320 J=J+1:IF J>K THEN 5200 ELSE 5220
7000 '*****
7001 '*' Transfer database data '*'
7002 '*****
7003 '
7010 IF ND THEN GOSUB 55700:GOTO 600
7015 CLS:GOSUB 54300:PRINT TAB(23)"-*- Transfer database data -*-":GOSUB 543
00:I$=""
7020 LOCATE 11,:PRINT"Which drive contains the source database?";
7030 L=1:T=2:UC=-1:GOSUB 52600:IF I$="" OR A=17 THEN 400
7040 IF I$<"A" OR I$>"Z" THEN GOSUB 55100:GOTO 7020
7050 GOSUB 54500:TW=11:BW=11:GOSUB 54400:D$=I$:GOSUB 50800
7060 IF E=0 THEN 7100
7070 IF E=18 THEN LOCATE 11,24:PRINT"No databases found on this drive":GOTO 7090
7080 LOCATE 11,24:PRINT"Error in reading disk directory!"
7090 LOCATE 13,23:PRINT"Hit any key to return to the menu":GOSUB 52000:GOTO 400
7100 XD=1
7110 LOCATE 5,1,0:PRINT" Use the up and down arrows to scan through the availa
ble databases, Q or control Q to quit."
7120 PRINT"When the desired source database is found, press the space bar."
7130 COLOR FH,BD:GOTO 7220
7140 GOSUB 52000:IF A=32 THEN 7230
7150 IF A=17 OR A=81 THEN 400
7160 IF SC=72 THEN 7180
7170 IF SC=80 THEN 7200 ELSE 7140
7180 XD=XD-1:IF XD<1 THEN XD=N
7190 GOTO 7210
7200 XD=XD+1:IF XD>N THEN XD=1
7210 LOCATE 11,1,0:PRINT LEFT$(SP$,80);
7220 LOCATE 11,(80-LEN(N$(XD)))/2,0:PRINT N$(XD);:GOTO 7140
7230 DD$=CD$:CD$=D$+"":N$(XD)
7240 COLOR FD,BD:TW=4:BW=11:GOSUB 54400
7250 LOCATE 11,13:PRINT>Loading source database information....Please wait...."
7260 FOR XF=1 TO MF:T2(XF)=T(XF):L2(XF)=L(XF):NEXT:TF=MF:XR=RL
7300 OPEN"R",4,CD$+".MSC",2:FIELD 4,2 AS M2$
7310 GET 4,1:RL=CVI(M2$):GET 4,2:MF=CVI(M2$):GET 4,4:TR=CVI(M2$):CLOSE 4
7320 OPEN"R",4,CD$+".FLD",20:FIELD 4,1 AS FL$,16 AS F$,1 AS T$,1 AS L$,1 AS UC$
7330 FOR XF=1 TO MF:GET 4,XF
7340 F2$(XF)=LEFT$(F$,ASC(FL$)):T(XF)=ASC(T$):L(XF)=ASC(L$)
7350 NEXT:CLOSE 4
7360 F=4:GOSUB 50000:RL=XR:CD$=DD$
7362 IF MF>TF THEN X=MF ELSE X=TF
7365 SWAP TF,MF:RL=XR:FOR XF=1 TO X:SWAP T2(XF),T(XF):L(XF)=L2(XF):NEXT
7367 IF TR=0 THEN GOSUB 56400:GOSUB 53100:GOTO 400
7370 TW=11:BW=11:GOSUB 54400
7400 LOCATE 5,1:PRINT" Use the up and down arrows to scan through the available
database fields,:PRINT"Q or Control Q to quit."
7410 PRINT" When the desired source field for the current destination field is
found,:PRINT"press the space bar. To skip the current source field, press the
ENTER key."
7420 LOCATE 11,5:PRINT"Source field":LOCATE 13,1:PRINT"Destination field":X=1:
X2=0:COLOR FH,BD:LOCATE 13,36:PRINT F$(X2+1);LEFT$(SP$,16);:GOTO 7510
7430 GOSUB 52000:GOSUB 54500:COLOR FH,BD:IF A$="Q" OR A=17 THEN CLOSE 3,4:GOTO 4
00
7440 IF A=32 THEN 7520
7450 IF A=13 THEN X=0:GOTO 7530
7460 IF SC=72 THEN 7480

```



```

7470 IF SC=80 THEN 7500 ELSE 7430
7480 X=X-1:IF X<1 THEN X=TF
7490 GOTO 7510
7500 X=X+1:IF X>TF THEN X=1
7510 LOCATE 11,35:PRINT F2$(X);LEFT$(SP$,16);:GOTO 7430
7520 IF (T(X2)>3 AND T(X2)<T2(X)) OR (T(X2)<4 AND T2(X)>3) THEN SL$="--*-- Sour
ce field does not match destination field's type --*--":GOSUB 54700:GOTO 7430
7530 X2=X2+1:SP(X2)=X:IF X2<MF THEN LOCATE 13,36:PRINT F$(X2+1);LEFT$(SP$,16);:G
OTO 7500
7540 IF X2<MF THEN LOCATE 13,36:PRINT F$(X2+1);LEFT$(SP$,16);:GOTO 7500
7550 TW=5:BW=13:GOSUB 54400:GOSUB 54500:LOCATE 12,22:PRINT"Transferring data....P
lease wait....";
7600 FOR XR=1 TO TR:GET 4,XR:GET 5,XR
7610 FOR XF=1 TO MF:LSET RD$(1,XF)=RD$(2,SP(XF)):LSET RI$(1)=RI$(2):NEXT
7620 MR=MR+1:PUT 2,MR:PUT 3,MR:NEXT:CLOSE 4,5:GOTO 300
65000 SAVE"DAMAN.BAS"

```

Listing 5:

```

400 ' *** Main menu for overlay
405 ' *** Current overlay = PRINT.BAS
409 KEY OFF:COLOR FD,BD:CLS:LOCATE 4,1,0
410 GOSUB 54300:PRINT TAB(22)"--*-- Print/Report functions --*--":GOSUB 54300:P
RINT
420 PRINT TAB(29)"C)hange current report"
430 PRINT TAB(29)"D)efine output driver"
440 PRINT TAB(29)"E)rase output driver"
445 PRINT TAB(29)"F)ont/pitch change"
450 PRINT TAB(29)"G)enerate reports"
460 PRINT TAB(29)"K)ill report"
470 PRINT TAB(29)"O)utput driver selection"
480 PRINT TAB(29)"P)rint reports"
490 PRINT TAB(29)"S)et find paramaters"
500 PRINT:PRINT TAB(29)"R)eturn to main menu"
510 PRINT TAB(29)"H)elp"
600 PRINT:LOCATE 21,1,0:GOSUB 54300:PRINT TAB(34)"Enter Choice:":GOSUB 52100:GO
SUB 54500
610 ON INSTR("CDEFGKOPSRH",A$) GOTO 1000,3000,5000,17000,7000,9000,11000,13000,1
5000,900,800
620 SOUND 4000,1:GOTO 600
800 ' *** Main Menu Help
810 F$="PRINT.HLP":GOSUB 50900:IF HE THEN 600 ELSE 400
900 ' *** Return to main menu
910 GOSUB 56000:RUN"MAIN"
1000 '*****
1001 '* Change current report *
1002 '*****
1003 '
1010 IF LR=0 OR ND THEN GOSUB 55700:GOTO 600
1020 CLS:GOSUB 54300:PRINT TAB(23)"--*-- Change current report --*--":GOSUB 5430
0:PRINT
1030 PRINT" Use the up and down arrows to scan through the available reports."
1040 PRINT"When the desired report is found, press the space bar."
1050 PRINT" To abort this function, press Q or control Q."
1060 X=0:COLOR FH,BD:GOTO 1130
1070 GOSUB 52000:IF A$="Q" OR A=17 THEN GOSUB 56700:GOSUB 53100:GOTO 400
1080 IF A=32 THEN 1160
1090 IF SC=72 THEN 1110

```



```

1100 IF SC=80 THEN 1130 ELSE 1070
1110 X=X-1:IF X<1 THEN X=LR
1120 GOTO 1140
1130 X=X+1:IF X>LR THEN X=1
1140 TW=11:BW=11:GOSUB 54400
1150 LOCATE 11,(80-LEN(RN$(X)))/2,0:PRINT RN$(X);:GOTO 1070
1160 COLOR FD,BD:CR=X:CR$=RIGHT$(ZF$+HEX$(CR),2):NP=0
1170 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE 4
1180 GOTO 250
3000 *****
3001 *                               Define output driver                               *
3002 *****
3003 '
3010 CLS:GOSUB 54300:PRINT TAB(24)"-=*= Define output driver -=*=":GOSUB 54300
:PRINT
3020 LOCATE 11,1:PRINT"Output driver name:";
3030 L=60:T=1:UC=0:GOSUB 52600:IF L=0 OR A=17 THEN 400
3040 OD$(MD+1)=I$:I$="LPT1:"
3050 LOCATE 13,1:PRINT"Output driver device name (? to prompt on print):";
3060 L=12:T=1:UC=-1:GOSUB 52600:IF A=17 THEN 400 ELSE IF L=0 THEN GOSUB 55200:GO
TO 3050
3090 DN$=I$:MD=MD+1:PM=0
3100 PM=PM+1:GOSUB 54500:TW=8:BW=14:GOSUB 54400:I$=""
3110 LOCATE 8,1:PRINT"Font/pitch";PM;"name:";
3120 L=60:T=1:UC=0:GOSUB 52600:IF L=0 OR A<13 THEN GOSUB 55200:GOTO 3110
3130 PN$(PM)=I$:I$="":GOSUB 54500
3140 NC=0
3150 NC=NC+1:I$=""
3160 LOCATE 10,1:PRINT"Control code";NC;"for font/pitch";PM;"(decimal ASCII code
)";
3170 L=3:T=3:GOSUB 52600:IF A=17 THEN SOUND 4000,1:GOTO 3140
3180 X=VAL(I$):IF L=0 OR X<0 OR X>255 THEN GOSUB 55200:GOTO 3160
3190 PC(PM,NC)=X:GOSUB 54500
3200 LOCATE 12,13:PRINT"Are there any more control codes for this font/pitch?";
3210 GOSUB 52100:IF A$="Y" THEN TW=12:BW=12:GOSUB 54400:GOTO 3150
3220 IF A$<"N" THEN GOSUB 56200:GOTO 3200
3230 NC(PM)=NC:GOSUB 54500
3240 LOCATE 14,23:PRINT"Are there any more font/pitches?";
3250 GOSUB 52100:IF A$="Y" THEN 3100
3260 IF A$<"N" THEN GOSUB 56200:GOTO 3240
3300 OD=MD:OD$=RIGHT$(ZF$+HEX$(OD),2)
3305 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE #4
3310 OPEN"O",4,"PRINTER.PTR":FOR X2=1 TO MD:PRINT #4,OD$(X2):NEXT:CLOSE 4:GOTO 1
7000
5000 *****
5001 *                               Erase output driver                               *
5002 *****
5003 '
5010 IF MD=0 THEN GOSUB 56900:GOTO 600
5020 CLS:GOSUB 54300:PRINT TAB(24)"-=*= Erase output driver -=*=":GOSUB 54300:
PRINT
5030 PRINT"    Use the up and down arrows to scan through the available printer d
rivers."
5040 PRINT"When the desired printer driver is found, press the space bar."
5050 PRINT"    To abort this function, press Q or control Q."
5060 X=0:COLOR FH,BD:GOTO 5130
5070 GOSUB 52000:IF A$="Q" OR A=17 THEN 400

```



```

5080 IF A=32 THEN 5160
5090 IF SC=72 THEN 5110
5100 IF SC=80 THEN 5130 ELSE 5070
5110 X=X-1:IF X<1 THEN X=MD
5120 GOTO 5140
5130 X=X+1:IF X>MD THEN X=1
5140 TW=11:BW=11:GOSUB 54400
5150 LOCATE 11,(80-LEN(SN$(X)))/2,0:PRINT SN$(X);:GOTO 5070
5160 LOCATE 14,14,0:PRINT"Are you sure you want to delete this printer driver?";
5170 GOSUB 52100:IF A$="N" THEN TW=14:BW=14:GOSUB 54400:GOTO 5070
5180 IF A$<>"Y" THEN GOSUB 56200:GOTO 5160
5190 X$=RIGHT$(ZF$+HEX$(X),2)
5200 KILL "DRIVER.D"+X$:MD=MD-1:IF X>MD THEN 5230
5210 FOR XS=X TO MD:X$=RIGHT$(ZF$+HEX$(XS),2):X2$=RIGHT$(ZF$+HEX$(XS+1),2)
5220 NAME "DRIVER.D"+X2$ AS "DRIVER.D"+X$:OD$(XS)=OD$(XS+1):NEXT
5230 OPEN"O",4,"PRINTER.PTR":FOR X2=1 TO MD:PRINT #4,OD$(X2):NEXT:CLOSE 4
5240 IF OD>X THEN OD=OD-1:GOTO 5270
5250 IF OD=X THEN OD=0:OD=-1:GOTO 5270
5260 GOTO 300
5270 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE #4
5280 IF NOT NS THEN 300
5290 TW=4:BW=14:GOSUB 54400
5300 LOCATE 11,3:PRINT"Since you have deleted the current driver, make sure to s
elect a new one!"
5310 LOCATE 13,25:PRINT"Hit any key to return to menu":GOSUB 52000:GOTO 300
7000 *****
7001 * Generate reports *
7002 *****
7003 '
7010 CLS:GOSUB 54300:PRINT TAB(26)"-=*= Generate reports -=*=":GOSUB 54300
7020 LOCATE 10,1,0:PRINT"Report Name:";
7030 I$="":L=66:T=1:UC=0:GOSUB 52600:IF I$="" OR A=17 THEN SOUND 4000,1:GOTO 400
7040 LR=LR+1:RN$(LR)=I$:I$="":TW=9:BW=11:GOSUB 54400:E=0
7050 LOCATE 6,2,0:PRINT"Report width (number of characters):";
7060 L=3:T=3:GOSUB 52600:GOSUB 54500:RW=VAL(I$):IF RW<1 OR RW>255 THEN GOSUB 552
00:GOTO 7050 ELSE I$=""
7070 LOCATE 8,5,0:PRINT"Number of lines per printed page:";
7080 L=3:T=3:GOSUB 52600:GOSUB 54500:PL=VAL(I$):IF PL<1 THEN GOSUB 55200:GOTO 70
70 ELSE I$=""
7090 LOCATE 10,7,0:PRINT"Number of lines for top margin:";
7100 L=2:T=3:GOSUB 52600:GOSUB 54500:TM=VAL(I$):IF TM<0 THEN GOSUB 55200:GOTO 70
90 ELSE I$=""
7105 IF TM=PL THEN GOSUB 56800:GOTO 7090
7110 LOCATE 12,4,0:PRINT"Number of lines for bottom margin:";
7120 L=2:T=3:GOSUB 52600:GOSUB 54500:BM=VAL(I$):IF BM<0 THEN GOSUB 55200:GOTO 71
10 ELSE I$=""
7125 IF BM=PL OR TM+BM=PL THEN GOSUB 56800:GOTO 7110
7130 LOCATE 14,1,0:PRINT"Number of characters for left margin:";
7140 L=2:T=3:GOSUB 52600:GOSUB 54500:LM=VAL(I$):IF LM<0 THEN GOSUB 55200:GOTO 71
30
7145 IF LM=RW THEN GOSUB 56800:GOTO 7130
7150 LOCATE 16,23:PRINT"Is the above information correct?";:GOSUB 52100
7160 IF A$="N" THEN TW=6:BW=16:GOSUB 54400:GOTO 7050
7170 IF A$<>"Y" THEN GOSUB 56200:GOTO 7150
7180 I$="":TW=6:BW=16:GOSUB 54400:GOSUB 54500:E=0
7300 LOCATE 10,2:PRINT"On the next page, enter the report definition using the f
ull screen editor."
7310 PRINT TAB(3)"To exit the full screen editor, press CTRL Q. Press any key t

```



```

o continue.":GOSUB 52000
7315 GOSUB 54500
7320 GOSUB 53300:GOSUB 54500
7330 LOCATE 25,16:PRINT"Are you sure you are done creating the report?";:GOSUB 5
2100
7340 IF A$="N" THEN E=-1:GOTO 7320
7345 IF A$<"Y" THEN GOSUB 56200:GOSUB 53100:GOTO 7330
7350 GOSUB 54500:LOCATE 25,1:PRINT"Position highlight over start of field block.
Hit the space bar to select.":X2=1:X=1:GOSUB 7410:BS=X
7355 GOSUB 54500:LOCATE 25,1:PRINT"Position highlight over end of field block.
Hit the space bar to select.":GOSUB 7410:BB=X
7360 IF BS>BB THEN SL$="-*=- End must be after start of field block -*=-":GOSU
B 54700:GOSUB 53100:GOTO 7350
7365 COLOR FE,BE:FOR X=BS TO BB:LOCATE X,1:PRINT MID$(RS$(X),SP-XC+1,80);:NEXT:C
OLOR FD,BD
7370 GOSUB 54500:LOCATE 25,10:PRINT"Is the highlighted area the desired repetati
ve field block?";:GOSUB 52100
7375 IF A$="N" THEN FOR X=BS TO BB:LOCATE X,1:PRINT MID$(RS$(X),SP-XC+1,80);:NEX
T:GOTO 7350
7376 IF A$<"Y" THEN GOSUB 56200:GOSUB 53100:GOTO 7370 ELSE 7500
7380 GOSUB 52000:IF A=32 THEN RETURN
7385 IF SC=72 THEN 7395
7390 IF SC=80 THEN 7405 ELSE 7380
7395 X=X-1:IF X<1 THEN X=24
7400 GOTO 7410
7405 X=X+1:IF X>24 THEN X=1
7410 GOSUB 7415:LOCATE X,1:COLOR FE,BE:PRINT MID$(RS$(X),SP-XC+1,80);:X2=X:GOTO
7380
7415 LOCATE X2,1:COLOR FD,BD:PRINT MID$(RS$(X2),SP-XC+1,80);:RETURN
7500 CLS:COLOR FD,BD:GOSUB 54500:LOCATE 10,23:PRINT"Compiling report...Please Wa
it..."
7510 GOSUB 53500:IF RT=0 THEN GOSUB 57000:E=-1:GOTO 7320
7515 IF LL>PL-TM-BM THEN GOSUB 57100:CLS:E=-1:GOTO 7050
7520 XF=1:XR=0:FOR XS=1 TO RT:CLS:COLOR FH,BD:X2=XR(XS)-2:IF X2<1 THEN X2=1
7530 FOR X3=X2 TO XR(XS):PRINT MID$(RS$(X3),YR(XS),80):NEXT:LOCATE CSRLIN-1,1
7540 COLOR FE,BE:PRINT MID$(RS$(XR(XS)),YR(XS),1):COLOR FD,BD:GOSUB 54300
7550 PRINT" Use the up and down arrows to scan through the available database
fields."
7560 PRINT"When the desired field is found, press the space bar. To skip a fiel
d, press":PRINT"the ENTER key."
7570 COLOR FH,BD:GOTO 7660
7580 GOSUB 52000:IF A=32 THEN 7670
7590 IF A=13 THEN 7720
7600 IF SC=72 THEN 7620
7610 IF SC=80 THEN 7640 ELSE 7580
7620 XF=XF-1:IF XF<1 THEN XF=MF+9
7630 GOTO 7650
7640 XF=XF+1:IF XF>MF+9 THEN XF=1
7650 LOCATE 11,1,0:PRINT LEFT$(SP$,80);
7660 LOCATE 11,(80-LEN(F$(XF)))/2,0:PRINT F$(XF);:GOTO 7580
7670 COLOR FD,BD:IF (RT(XS)=1 AND (T(XF)<5 OR T(XF)=9)) OR (RT(XS)=2 AND T(XF)>4
AND T(XF)<8) THEN 7710
7680 LOCATE 15,7,0:PRINT"Field types do not match, has the correct field been se
lected?";
7690 GOSUB 52100:IF A$="N" THEN TW=15:BW=15:GOSUB 54400:GOTO 7580
7700 IF A$="Y" THEN E=-1:GOTO 7320 ELSE GOSUB 56200:GOTO 7680
7710 XR=XR+1:RF(XR)=XF:RY(XR)=YR(XS):RX(XR)=XR(XS)
7720 NEXT:IF XR=0 THEN GOSUB 57000:E=-1:GOTO 7320
7730 RM=XR:NP=0:CR=LR:CR$=RIGHT$(ZF$+HEX$(CR),2)

```



```

7740 OPEN"R",4,CD$+".R"+CR$,4:FIELD 4,2 AS SF$,1 AS PX$,1 AS PY$
7750 LSET SF$=MKI$(RM):LSET PX$=CHR$(RW):LSET PY$=CHR$(PL):PUT 4,1
7760 LSET SF$=CHR$(TM)+CHR$(BM):LSET PX$=CHR$(LM):LSET PY$=CHR$(LL):PUT 4,2
7770 LSET PX$=CHR$(BS):LSET PY$=CHR$(BB):PUT 4,3
7780 FOR XS=1 TO RM:LSET SF$=MKI$(RF(XS)):LSET PX$=CHR$(RX(XS))
7790 LSET PY$=CHR$(RY(XS)):PUT 4:NEXT:CLOSE 4
7800 OPEN"R",4,CD$+".P"+CR$,RW:FIELD 4,RW AS S$
7810 FOR X=1 TO LL:LSET S$=RS$(X):PUT 4,X:NEXT:CLOSE 4
7820 OPEN"O",4,CD$+".RPT":FOR X=1 TO LR:PRINT #4,RN$(X):NEXT:CLOSE 4
7830 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE 4
7840 LSET MX$=CHR$(LD)+CHR$(LR):PUT 1,5:GOTO 300
9000 *****
9001 '*                      Kill report                      *
9002 *****
9003 '
9010 IF NP OR ND THEN GOSUB 55700:GOTO 600
9020 CLS:GOSUB 54300:PRINT TAB(28)"-=-*- Kill report -=-*-":GOSUB 54300:PRINT
9030 PRINT"    Use the up and down arrows to scan through the available reports."
9040 PRINT"When the desired report is found, press the space bar."
9050 PRINT"    To abort this function, press Q or control Q."
9060 X=0:COLOR FH,BD:GOTO 9160
9100 GOSUB 52000:IF A$="Q" OR A=17 THEN 400
9110 IF A=32 THEN 9190
9120 IF SC=72 THEN 9140
9130 IF SC=80 THEN 9160 ELSE 9100
9140 X=X-1:IF X<1 THEN X=LR
9150 GOTO 9170
9160 X=X+1:IF X>LR THEN X=1
9170 TW=11:BW=11:GOSUB 54400
9180 LOCATE 11,(80-LEN(RN$(X)))/2,0:PRINT RN$(X);:GOTO 9100
9190 LOCATE 14,17,0:PRINT"Are you sure you want to delete this report?";
9200 GOSUB 52100:IF A$="N" THEN TW=14:BW=14:GOSUB 54400:GOTO 9100
9210 IF A$<"Y" THEN GOSUB 56200:GOTO 9190
9220 X$=RIGHT$(ZF$+HEX$(X),2)
9230 KILL CD$+".R"+X$:KILL CD$+".P"+X$:LR=LR-1:IF X>LR THEN 9300
9240 FOR XS=X TO LR:X$=RIGHT$(ZF$+HEX$(XS),2):X2$=RIGHT$(ZF$+HEX$(XS+1),2)
9250 NAME CD$+".R"+X2$ AS CD$+".R"+X$
9260 NAME CD$+".P"+X2$ AS CD$+".P"+X$:RN$(XS)=RN$(XS+1):NEXT
9300 LSET MX$=CHR$(LD)+CHR$(LR):PUT 1,5
9310 OPEN"O",4,CD$+".RPT":FOR X2=1 TO LR:PRINT #4,RN$(X2):NEXT:CLOSE 4
9320 IF CR>X THEN CR=CR-1:GOTO 9350
9330 IF CR=X THEN CR=0:NP=-1:GOTO 9350
9340 GOTO 300
9350 OPEN"O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLO
SE #4
9450 IF NOT NP THEN 300
9460 TW=4:BW=14:GOSUB 54400
9470 LOCATE 11,4:PRINT"Since you have deleted the current report, make sure to s
elect a new one!"
9480 LOCATE 13,25:PRINT"Hit any key to return to menu":GOSUB 52000:GOTO 300
11000 *****
11001 '*                      Output driver selection                      *
11002 *****
11003 '
11010 IF MD=0 THEN GOSUB 56900:GOTO 600
11020 CLS:GOSUB 54300:PRINT TAB(22)"-=-*- Output driver selection -=-*-":GOSUB 5
4300:PRINT
11030 PRINT"    Use the up and down arrows to scan through the available output d

```



```

rivers."
11040 PRINT "When the desired output driver is found, press the space bar."
11050 PRINT " To abort this function, press Q or control Q."
11060 X=0:COLOR FH,BD:GOTO 11130
11070 GOSUB 52000:IF A$="Q" OR A=17 THEN GOSUB 56700:GOSUB 53100:GOTO 400
11080 IF A=32 THEN 11160
11090 IF SC=72 THEN 11110
11100 IF SC=80 THEN 11130 ELSE 11070
11110 X=X-1:IF X<1 THEN X=MD
11120 GOTO 11140
11130 X=X+1:IF X>MD THEN X=1
11140 TW=11:BW=11:GOSUB 54400
11150 LOCATE 11,(80-LEN(OD$(X)))/2,0:PRINT OD$(X);:GOTO 11070
11160 COLOR FD,BD:OD=X:OD$=RIGHT$(ZFS+HEX$(OD),2):NO=0
11170 OPEN "O",4,"CDBASE.PTR":PRINT #4,CD$:PRINT #4,SN:PRINT #4,CR:PRINT #4,OD:CLOSE 4
11180 OPEN "I",4,"DRIVER.D"+OD$:INPUT #4,CP,PM,DN$
11190 FOR X=1 TO PM:INPUT #4,PN$(X),NC(X)
11200 FOR XC=1 TO NC(X):INPUT #4,PC(X,XC):NEXT:CLOSE 4:GOTO 400
130000 '*****
130001 '* Print reports *
130002 '*****
130003 '
13010 IF ND OR NP OR NO THEN GOSUB 55700:GOTO 600 ELSE IF MR<1 THEN GOSUB 56400:GOTO 600
13020 CLS:GOSUB 54300:PRINT TAB(27)"--*-- Print reports --*--":GOSUB 54300:PRINT
13025 PP=0:FM=0:TS=0:NM=0:EM=0:IF QS=0 THEN 13060
13030 I$="N":LOCATE 6,3:PRINT "Do you want to print by find paramaters?";
13040 L=1:T=2:UC=-1:GOSUB 52600:IF I$="" OR I$="Q" OR A=17 THEN 400 ELSE IF I$="Y" THEN FM=-1:GOTO 13060
13050 IF I$<"N" THEN GOSUB 56200:GOTO 13030
13060 I$="N":LOCATE 8,8:PRINT "Do you want to print by tag status?";
13070 L=1:T=2:UC=-1:GOSUB 52600:GOSUB 54500:IF I$="Y" THEN TS=-1:GOTO 13090
13080 IF I$<"N" THEN GOSUB 56200:GOTO 13060
13090 I$="N":LOCATE 10,1:PRINT "Do you want to print by new record status?";
13100 L=1:T=2:UC=-1:GOSUB 52600:GOSUB 54500:IF I$="Y" THEN NM=-1:GOTO 13120
13110 IF I$<"N" THEN GOSUB 56200:GOTO 13090
13120 I$="N":LOCATE 12,7:PRINT "Do you want to print by edit status?";
13130 L=1:T=2:UC=-1:GOSUB 52600:GOSUB 54500:IF I$="Y" THEN EM=-1:GOTO 13150
13140 IF I$<"N" THEN GOSUB 56200:GOTO 13120
13150 I$="0":LOCATE 14,20:PRINT "Access level of report:";
13160 L=2:T=3:GOSUB 52600:GOSUB 54500:AL=VAL(I$)
13170 IF AL<0 OR AL>31 THEN GOSUB 55200:GOTO 13150
13180 I$="N":LOCATE 16,8:PRINT "Do you wish to pause between pages?";
13190 L=1:T=2:UC=-1:GOSUB 52600:GOSUB 54500:IF I$="Y" THEN PP=-1:GOTO 13210
13200 IF I$<"N" THEN GOSUB 56200:GOTO 13180
13210 XD$=DN$:IF LEFT$(DN$,1)<"?" THEN 13300 ELSE IF LEN(DN$)>1 THEN I$=MID$(DN$,2) ELSE I$=""
13220 LOCATE 18,1:PRINT "Output device/filename:";
13230 L=12:T=1:UC=-1:GOSUB 52600:GOSUB 54500:IF I$="" THEN GOSUB 55200:GOTO 13220 ELSE XD$=I$
13300 IF XD$="SCRN:" THEN CLS:GOTO 13500
13310 TW=5:BW=18:GOSUB 54400:LOCATE 12,25:PRINT "Hit any key to start printing";:GOSUB 52000
13320 TW=12:BW=12:GOSUB 54400:LOCATE 12,36:PRINT "Printing"
13500 OPEN "O",4,XD$:BP=-1:PA=PL-TM-BM:RS=BB-BS+1:ES=LL-BB:PN=0:LN=0:RD$(1,MF+8)=TIME$:RD$(1,MF+9)=DATE$
13600 FOR X=1 TO NC(CP):PRINT #4,CHR$(PC(CP,X));:NEXT
14000 FOR RN=1 TO MR:GOSUB 50400:IF AL<DR THEN 14800

```



```

14010 IF TS AND TG=0 THEN 14800
14020 IF EM AND ED=0 THEN 14800
14030 IF NM AND NR=0 THEN 14800
14040 IF NOT FM THEN GET 2,RN:GOTO 14100
14050 GOSUB 50600:IF NOT DF THEN 14800
14100 LN=LN+1:RD$(1,MF+3)=MKIS(LN):RD$(1,MF+2)=MKIS(RN):RD$(1,MF+4)=MKIS(DR)
14101 IF TG=0 THEN RD$(1,MF+5)=" " ELSE RD$(1,MF+5)="*"
14102 IF ED=0 THEN RD$(1,MF+6)=" " ELSE RD$(1,MF+6)="*"
14103 IF NR=0 THEN RD$(1,MF+7)=" " ELSE RD$(1,MF+7)="*"
14105 IF NOT BP THEN 14200
14110 PN=PN+1:RD$(1,MF+1)=MKIS(PN):BP=0:FS=1:PR=BS-1
14115 GOSUB 14850
14120 PRINT #4,STRING$(TM,13);
14130 IF BS>1 THEN XS=1:XE=BS-1:XF=1:GOSUB 14900:FS=XF
14200 XS=BS:XE=BB:XF=FS:GOSUB 14900
14210 PR=PR+RS:IF PR+RS+ES<=PA THEN 14800
14300 IF BB=LL THEN 14400
14310 XS=BB+1:XE=LL:GOSUB 14900
14400 PRINT #4,STRING$(PL-PR-ES-TM,13);:BP=-1
14800 NEXT:IF BB=LL OR BP THEN 14820
14810 XS=BB+1:XE=LL:GOSUB 14900
14820 GOSUB 14850:CLOSE 4:GOTO 400
14850 IF PP AND PN>1 THEN SL$="Hit any key to continue, Q or control Q to quit":
SOUND 4000,1:GOSUB 54600:GOSUB 52000:IF A=17 OR A$="Q" THEN CLOSE 4:RETURN 400 E
LSE GOTO 54500 ELSE RETURN
14900 ' *** Print report from RS$(XS) to RS$(XE)
14910 FOR XL=XS TO XE:XC=1
14915 IF XL<RX(XF) OR XF>RM THEN PRINT #4,LEFT$(SP$,LM);RS$(XL);:GOTO 14995
14920 X=RY(XF+1):IF RX(XF+1)<>RX(XF) OR XF=RM THEN X=RW
14925 FD$=MID$(RS$(XL),XC,X-XC):ON T(RF(XF)) GOTO 14930,14930,14930,14932,14935,
14940,14945,14932
14930 PRINT #4,USING FD$;RD$(1,RF(XF));:GOTO 14950
14932 XY=CVI(LEFT$(RD$(1,RF(1,XF)),2)):XM=ASC(MID$(RD$(1,RF(1,XF)),3,1)):XD=ASC(
RIGHT$(RD$(1,RF(1,XF)),1)):PRINT #4,USING FD$;RIGHT$(ZF$+MID$(STR$(XM),2),2);"-";
RIGHT$(ZF$+MID$(STR$(XD),2),2);"-";RIGHT$(ZF$+MID$(STR$(XY),2),4):GOTO 14950
14935 PRINT #4,USING FD$;CVI(RD$(1,RF(XF)));:GOTO 14950
14940 PRINT #4,USING FD$;CVS(RD$(1,RF(XF)));:GOTO 14950
14945 PRINT #4,USING FD$;CVD(RD$(1,RF(XF)));:GOTO 14950
14950 XC=X:X=XF+1:IF XC<RW THEN GOTO 14920
14995 PRINT #4,:NEXT:RETURN
15000 '*****
15001 '* Set find paramaters *
15002 '*****
15003 '
15010 IF ND THEN GOSUB 55700:GOTO 600
15020 QS=0:I$=""
15030 CLS:GOSUB 54300:PRINT TAB(24)"-=*=- Set find paramaters -=*=-":GOSUB 54300
:PRINT:GOSUB 54500
15040 QS=QS+1
15050 LOCATE 5,1,0:PRINT" Use the up and down arrows to scan through the avail
able reports."
15060 PRINT"When the desired search field is found, press the space bar."
15070 PRINT" To abort this function, press Q or control Q."
15080 XF=1:COLOR FH,BD:GOTO 15160
15090 GOSUB 52000:IF A$="Q" OR A=17 THEN 400
15100 IF A=32 THEN 15180
15110 IF SC=72 THEN 15130
15120 IF SC=80 THEN 15150 ELSE 15090
15130 XF=XF-1:IF XF<1 THEN XF=MF

```



```

15140 GOTO 15160
15150 XF=XF+1:IF XF>MF THEN XF=1
15160 TW=11:BW=11:GOSUB 54400
15170 LOCATE 11,(80-LEN(F$(XF)))/2,0:PRINT F$(XF);:GOTO 15090
15180 IS="":QN(QS)=XF:T=T(XF):IF T>4 THEN T=3 ELSE IF T>2 THEN T=1
15190 LOCATE 13,1:PRINT"Data to search for:";
15200 L=L(XF):IF L>60 THEN L=60
15210 UC=UC(XF):GOSUB 52600:GOSUB 54500:IS="":ON T(XF) GOTO 15220,15220,15220,15
230,15240,15250,15260,15230
15220 ST$(QS)=A$:GOTO 15270
15230 IF L<10 THEN GOSUB 55200:GOTO 15190 ELSE A$=MKIS(VAL(RIGHT$(A$,4)))+CHR$(
VAL(LEFT$(A$,2)))+CHR$(VAL(MID$(A$,4,2))):ST$(QS)=MID$(A$,2,1)+LEFT$(A$,1)+RIGHT
$(A$,2):GOTO 15270
15240 ST$(QS)=MKIS(VAL(A$)):GOTO 15270
15250 ST$(QS)=MKSS(VAL(A$)):GOTO 15270
15260 ST$(QS)=MKDS(VAL(A$))
15270 LOCATE 15,1:PRINT"Relation between search data and database data:"
15280 PRINT,"1) Absolute equivalent","4) Less than"
15290 PRINT,"2) Left part equivalent","5) Greater than"
15300 PRINT,"3) Instring equivalent"
15310 LOCATE 20,25:PRINT"What is the relation number?";
15320 L=1:T=3:GOSUB 52600:GOSUB 54500:X=VAL(IS):IF X<1 OR X>5 THEN GOSUB 55200:G
OTO 15310
15330 RE(QS)=X:IF X=1 AND T(XF)<3 THEN ST$(QS)=ST$(QS)+LEFT$(SP$,L(XF)-LEN(ST$(Q
S)))
15340 LOCATE 22,9:PRINT"Have the search paramaters for this field been set corre
ctly?";
15350 GOSUB 52100:GOSUB 54500:IF A$="N" THEN 15050 ELSE IF A$<"Y" THEN GOSUB 56
200:GOTO 15340
15360 LOCATE 24,17:PRINT"Are there any more paramaters to this search?";
15370 GOSUB 52100:GOSUB 54500:IF A$="Y" THEN 15040 ELSE IF A$<"N" THEN GOSUB 56
200:GOTO 15360
15380 GOTO 400
17000 '*****
17001 '* Font/pitch change *
17002 '*****
17003 '
17010 IF PM=0 THEN GOSUB 56900:GOTO 600
17020 CLS:GOSUB 54300:PRINT TAB(25)"--*-- Font/pitch change --*--":GOSUB 54300:P
RINT
17030 PRINT" Use the up and down arrows to scan through the available output d
rivers."
17040 PRINT"When the desired font/pitch is found, press the space bar."
17050 PRINT" To abort this function, press Q or control Q."
17060 X=0:COLOR FH,BD:GOTO 17130
17070 GOSUB 52000:IF A$="Q" OR A=17 THEN GOSUB 56700:GOSUB 53100:GOTO 400
17080 IF A=32 THEN 17160
17090 IF SC=72 THEN 17110
17100 IF SC=80 THEN 17130 ELSE 17070
17110 X=X-1:IF X<1 THEN X=PM
17120 GOTO 17140
17130 X=X+1:IF X>PM THEN X=1
17140 TW=11:BW=11:GOSUB 54400
17150 LOCATE 11,(80-LEN(PN$(X)))/2,0:PRINT PN$(X);:GOTO 17070
17160 CP=X:COLOR FD,BD:OPEN"O",4,"DRIVER.D"+OD$:PRINT #4,CP:PRINT #4,PM:PRINT #4
,DN$
17170 FOR X=1 TO PM:PRINT #4,PN$(X):PRINT #4,NC(X)
17180 FOR XC=1 TO NC(X):PRINT #4,PC(X,XC):NEXT:NEXT:CLOSE 4:GOTO 400
65000 SAVE"PRINT.BAS"

```


Pop-Up DeskSet

A Great Organizer

Software 1000/1200/2000/3000

For years now computers have been doing everything from simple math to putting a man on the moon. Surely all of this vast technology could be used to somehow organize my desk! Well, now you can go to the moon and organize your desk too.

Recently, there have been a rash of new desk-top organizers appearing on the market. A great deal of them are for IBM compatible computers such as the Tandy 1000 and 1200.

One to recently appear on the scene is *Pop-Up DeskSet* and *Pop-Up DeskSet Plus*. The two packages are identical except the *Plus* version includes the module *Pop-Up Modem* for data communications as well as the standard module of *Pop-Up Voice* for voice communications. I will restrict my review to *Pop-Up DeskSet*.

Pop-Up DeskSet, by Bellsoft, Inc., is a complete set of desk-top organization tools that are available at the touch of a key — even while running another program. These tools include a full-function calculator, business calculator, notepad, clipboard, calendar, alarm clock, voice dialer, DOS functions and an amazing *Pop-up Anything*. This last item is the best thing since pop-up pastries.

The package comes in an attractive

folding plastic box and includes three paperback manuals, two quick reference cards and a single diskette. The instructions are clearly presented with only a few very minor typing errors in the manual. Two of these appear on Page 16 of the main manual. The first is in Section 3, Paragraph 2 stating “simply press ALT-N and the key that stands for the desk tool.” This should say “simply press ALT and the key that stands for the desk tool.” The second is in Section 5, Paragraph 1 stating “and the ALT-C will be passed on to your application.” This should read “and the ALT sequence will be passed on to your application.”

System requirements for running *Pop-Up DeskSet* are a Tandy 1000, 1200, 3000 or other IBM compatible, 64K RAM (to run *Pop-Up Voice*, you need 128K) and one disk drive. *Pop-Up Voice* also requires a Hayes compatible modem. Unlike some desk-top organizers, *DeskSet* allows you to select each individual module to suit your own needs and memory requirements. Each module is entirely RAM-resident and all help files are contained within the module itself. This means once *DeskSet* is loaded none of its modules are required to remain in any drive. The modules themselves require anywhere from 13K for the alarm to 31K for the calendar. *Pop-Up Anything* requires various amounts of RAM depending on the applications you are running. Since

each module works stand-alone, they may be placed on your application diskettes as you require.

The first thing you will want to do is create a working copy of *DeskSet*. You

A program called *TLCSETUP* allows you to configure *Pop-Up Voice* for your system. You are allowed to choose the serial port (COM1 or COM2), Touch Tone dialing, a PBX dial-out sequence and a

ALT-C *Pop-up Calculator*
ALT-X *Pop-up Financial Calculator*
ALT-D *Pop-up Calendar*
ALT-N *Pop-up Notepad*
ALT-U *Pop-up PopDos*
ALT-V *Pop-up Voice*
ALT-F10 *Pop-up Anything*

"The unique feature of the alarm clock function is its ability to send command strings to the computer at preset times."

are provided with a utility to create or update your *AUTOEXEC.BAT* file. Simply type *AUTOPOP* and answer the questions as they are presented to you. You may select any combination or all of the "pop-ups" if you wish. When you are finished, you will have a new *AUTOEXEC.BAT* file. When you boot your system using this disk, the selected pop-ups will automatically be loaded one at a time. *Pop-Up Alarm* should always be the first module loaded and *Pop-Up Anything* should be the last.

long distance service access and user code if you wish. You may also enter three commonly-used phone numbers if you like. If you have a color monitor, you may also set foreground and background colors.

Each Pop-Up module has its own unique invocation key. The following are the ones I am going to cover in the rest of this review:

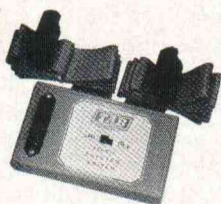
ALT-A *Pop-up Alarm Clock*
ALT-B *Pop-up Clipboard*

If these key combinations are used in your applications, simply use them twice. The second entry will turn off the pop-up and will be passed on to your application. The key sequence ALT-F may also be used to "feed" information from pop-up to pop-up or from pop-up to application.

There are four functions that are common to all of the pop-ups. You may position the pop-ups on the screen by using CTRL-right arrow to move clockwise or CTRL-left arrow to move counter-clockwise. To toggle the color on and off, use CTRL-C. The F1 key will invoke a help screen for all pop-ups. The remainder of the options for each pop-up appear on a menu bar to the right of each one. They may be chosen by using a function key assigned to each option or by using the + (plus) and -

WHY FIGHT IT? . . . SWITCH!

TABCO PARALLEL Printer Switch connect 2 printers to one computer! PSM-C/2P — Manual printer switch. Prolongs the life of your printer/interface connectors. Saves time & money! Use your existing printer cable to connect TAB SWITCH to your computer. Comes complete with two 3.5' cables to connect your printers. Straight-forward design makes it easy to set-up & use. 1 year limited warranty.



ORDER TODAY
Only \$69.95

Other models available. Dealer inquiries invited. \$2.50 Shipping/Handling. Florida residents add 5% sales tax.

Cables built-in
(optional lengths available)

Nothing else to buy
Fingertip control
Just flick your switch

See your dealer
or
Call 1-800-874-1551

TAB SALES CO.

704 W. Michigan Ave.

Pensacola, FL 32505

P.O. Box 8098

Ph: 904-438-6507

MEET THE WIZARD™

Quality / Price / Performance Guaranteed

Easy to use! **\$29.95** for any one package
9.95 for each additional

For TANDY 1000, 1200, 2000, 3000 and Compatibles

- 1. RAMDISK** — Transform your spare memory into superfast disk(s). You specify size and number.
* **NEW** * replace your old ramdisk software. Ours is BETTER. Automatic timed backup. Never again lose your ram-disk data.
- 2. UNDELETE Plus** — Recover deleted files, display sorted directory tree, rename directories, cross directory file search, quick sort directory display, change file attributes, and **more** (8 programs in all).
- 3. SNAPSHOT** — Instantly save and display up to 99 text screens. Great for debugging, Help screens, etc.
- 4. SHERLOCK** — Find lost files. Give Sherlock a clue (eg phrase or fragment of file contents) and files will be located in all directories.
** **NEW! Memory Resident, Menu Driven** **
- 5. GRAFIX** — Send graphics screen image to printer. You choose printout size.
- 6. FONTIX** — Set printer fonts, tabs, margins, etc. Automatically produce paginated printouts and structured Basic listings.

WIZARD Utilities
P.O. Box 714
Sudbury, MA 01776

800-622-4070
800-942-7317 in ILL.
\$3.00 ship & handling
in Mass add 5% tax

(minus) sign keys on the numeric keypad to move a highlight bar and then using CTRL-E to invoke the option.

Using the second method above seemed to be a problem at first on the Tandy 1000. The +/- keys and the backslash (\) key generate different keyboard scan codes from other IBM compatibles. However, installing the KEYCNVRT.SYS driver in the CONFIG.SYS file solved this problem. To use these keys you must have the NUM LOCK key on.

Pop-Up Alarm Clock

The first module we'll discuss will be the alarm clock. *Pop-Up Alarm* allows you to do any of the following:

- 1) View the time of day whenever you like.
- 2) Set alarm clock options from the alarm clock menu.
- 3) Set alarm clock options on the command line when you load *Alarm Clock* or create a "setup" file to load preset alarms and options automatically.
- 4) Time events with a stopwatch.
- 5) Set your computer to run pro-

grams when you're not there.

- 6) Feed the current time and date to your applications.

The options include choosing when to display the time and when to chime. You may display the time each minute, each hour, constantly or not at all. The chime may sound once each hour, chime each hour or never. When an alarm goes off, you may press S to "snooze" or R to simply reset it. These options may also be set as you load the alarm clock module. To do this you must include an option parameter with the normal POPALARM command. Option parameters take the form of POPALARM *option*=*x* where *option* may be 1, location; d, display; b, bell; a, alarm; or f, feed. With alarm and feed you may also specify a message to display or a command string to send to the computer.

The most unique feature of the alarm clock functions is its ability to send command strings to the computer at preset times. This differs from the alarm feature in that you are only allowed to set one "feed" time whereas you may set up to six "alarm" times, each with its

own message to be displayed. The command string may contain DOS commands, program names or anything you wish. All special characters such as the backslash (\) and ENTER key must be preceded with a backslash (\). On a Tandy 1000 the use of the backslash also requires the KEYCNVRT.SYS file to be loaded, however the NUM LOCK key need not be on as it must in order to use the +/- keys on the keypad. The timed feed function of the alarm clock is a very powerful function indeed.

Pop-Up Calendar

The next module to discuss is the calendar. This module allows you to:

- 1) Look at a three-month calendar display.
- 2) Look at a detailed one-month calendar.
- 3) Mark special events.
- 4) Keep track of cyclic events.
- 5) Save your important dates in a file.
- 6) Get a date file.
- 7) Print calendars.
- 8) Stamp the date on another application.

PORTABLE COMPUTER MEMORY EXPANSION

24K Expansion RAM 8K Expansion RAM

For the Model 200

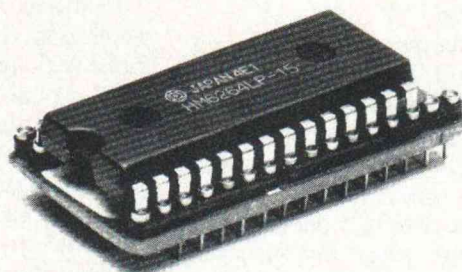
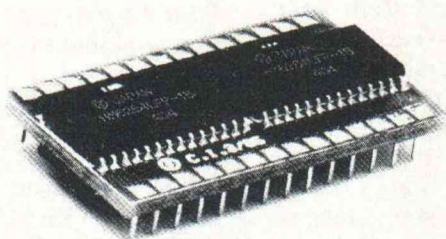
\$99. each

\$190. for two

For the Model 100, M 10 & 8201

\$27. each

\$75. set of 3



These modules are easily installable and feature the low power version CMOS static RAM. Illustrated step by step instructions with a memory test program listing included. Our aim is your satisfaction. There is a thirty day refund policy and a one year warranty. Phone and Mail orders are accepted and items are shipped promptly from stock. Call (714) 540-1174.

M/C, VISA, Check or Money Order

Shipping-Add \$1.50 UPS Ground
or \$4.00 2nd Day Air.

CA Residents add 6% Sales Tax



**American
Cryptronics, Inc.**

1580 Corporate Dr., Suite 123
Costa Mesa, CA 92626
Phone: (714) 540-1174

The calendar function is loaded by typing POPDATE and is activated with ALT-D. A three-month display appears showing the current month, the previous month and the following month. Using the cursor control keys, you may move forward or back through the months or through the years. You may view anywhere from 1900 to 2027. By using the F2 key, you will get a single-month display with the current date flashing. Up to two events per day will be shown where they have already been set. Pressing the M key will display the remainder of your messages if you have any.

One really nice feature of the calendar function is its ability to accept one event followed by a list of dates to track that event on. This is useful for events that do not always occur on the same date, such as Thanksgiving. For events occurring on the same day year-to-year there is a "cyclic" function whereby you specify an event, the day on which it occurs and the cycle (such as weekly or annually) that it occurs. This function is great for birthdays and anniversaries.

What if you have different people with different dates to remember? No problem! Using the "save" function you simply save your own dates in a file for later retrieval by the "get" function. A file named DATES.DAT is automatically loaded when you first load the calendar module but any file may be loaded after this. Using the calendar module you may also print a calendar or pass a date from the calendar to one of your other applications.

Pop-Up Notepad

The *Pop-Up Notepad* module is available in an instant. In fact, the notepad is where I kept my notes for writing this review while I was working with the various modules to see what they could do. Believe me, this is the only way to go. As you're reviewing a package you can pull up your notes any time you wish to jot down something about a particular feature or problem. *Pop-Up Notepad* has:

- 1) A "mini word processor" that lets you write and edit notes, lists and other text files.
- 2) Easy screen editing features.
- 3) Permanent files that can be stored, retrieved and edited at any time.
- 4) Help at the touch of a key.

To load the notepad, type ALT-N. A tall, narrow notepad-type window will

appear on the screen with a menu bar to the right of it.

The notepad allows you to enter text just as a word processor would. You can insert text and lines, delete text and lines, align paragraphs, store and retrieve notepad files and browse your directories for notepad files. Since the features of the notepad are a very small subset of a full word processor, I will not elaborate any further on them. Let me just say that the features that are there are more than sufficient to jot down notes or make phone lists to use with *Pop-Up Voice*.

Pop-Up Clipboard

The clipboard is similar to the notepad in many respects. It also is a mini word processor, however, the clipboard has a set of expanded block commands to assist in manipulating text. The clipboard takes up the full screen and in order to view the application underneath, you must use the F3 key. The Clipboard enables the user to do the following:

- 1) Enter and edit text.
- 2) Select and copy blocks of text from one part of the clipboard to another.
- 3) Copy blocks of text from the clipboard to an application, from an application to the clipboard or from the clipboard to another pop-up.
- 4) Adjust the speed at which the clipboard feeds text to another application.

Features 1 and 2 are simple word processing. Features 3 and 4 are what make the clipboard the powerful tool it is. Blocks are handled by first marking the beginning and ending and then copying, moving or deleting them. To copy from the clipboard to an application, you first place the cursor where you wish to copy text. Next, invoke clipboard with ALT-B and mark the block of text you wish to copy using the F5 key to begin the block and the cursor keys to highlight the block. Finally, type ALT-F to feed the text to your application. You will be asked for an end-of-line character. Type the character of ALT-F for none and the block will be copied to your application. To reverse the process, you enter the clipboard and use F3 to view your application. Mark the block to copy using F5 to begin, the cursor keys to select and F6 to end the block. You will be placed back into the

clipboard where you will see a flashing box. Place this box where you want to copy the text and type CTRL-I followed by any key to confirm copy or ESC to cancel. It's as simple as that.

If your application cannot accept text as fast as the clipboard sends it, you may adjust the feed speed when installing the clipboard by typing POPCLIP /rx where x is a digit from zero to nine, with zero being the fastest. Combining the clipboard with *Pop-Up Anything* allows you to copy text from one application to another.

Pop-Up Calculator

There are two calculators in the *DeskSet* package. The first is a simple four-function calculator with 10 memories and percent calculations. The second is a powerful financial calculator which even has its own separate manual of 37 pages. Both calculators provide a pictorial paper tape record of all calculations. This tape may be printed for a permanent record. Both also provide a help screen by using the F1 key. Pressing C clears the calculator while X clears only the current entry.

The financial calculator provides the following functions in addition to the standard four functions, percents and memories:

- 1) A powerful financial analysis tool that solves compound interest problems, loans, annuities, discounted cash flow analysis and more.
- 2) A statistical calculator that solves problems of both one and two variables.

The financial calculator has five different modes for performing calculations:

- 1) Normal
- 2) Interest
- 3) Annuity begin
- 4) Annuity end
- 5) Statistics

Typing CTRL-M toggles the different modes. CTRL-H brings up a large help screen which you may leave displayed as you perform calculations. This is a great help when you're dealing with as many variables as you have in finance and statistics. The 10 function keys are used as financial and statistical registers for values such as present value, future value, interest, payment, period, mean, sum of X, sum of Y and standard deviation.

The items that can be calculated with the financial calculator are so numerous I will only list a few. You can figure simple interest, compound interest, future value, effective interest rate, payment on annuity, annual percentage rate, amortization, depreciation, forecasting, slope and intercept, variable analysis and much more. Any value from either calculator may be placed in an application using the "feed" command.

Pop-Up Voice

Pop-Up Voice is used to autodial a number from your keyboard or from a phone list. You can:

- 1) Dial often-called numbers automatically with just a few keystrokes — even if you use a PBX number or a long distance phone service.
- 2) Dial phone numbers from your screen, phone list or database.

As was mentioned earlier, you must run *TLCSETUP* to customize the voice module. This lets you choose a serial port, three commonly called numbers, a PBX and long distance service number and tone dialing if you have it. Voice is displayed by ALT-V and will appear with your chosen numbers and any number brought forward from the cursor position when voice was invoked. This feature lets you create a phone list using the notepad or some other database program, display your list and then *Pop-Up Voice* pulling forward any of the numbers in the list. After dialing you simply pick up the phone. Remember, the *DeskSet Plus* version also adds data communications.

Pop-Up DOS

Pop-Up DOS gives you access to several DOS features while you are running another application. The features provided are:

- 1) DIR Disk directory.
- 2) XDIR See the first 20 characters in each file of a directory.
- 3) CD Change your current directory from within an application.
- 4) DEL Delete a file.
- 5) COPY Copy a file.
- 6) REN Rename a file.
- 7) TYPE Look at the contents of a file.
- 8) CHKD Allows you to run the DOS CHKDSK command.

In addition to these features the DOS module allows you to:

- 1) Print files in a choice of typefaces while you are working on something else.
- 2) Use your keyboard like a typewriter to print whatever you like.

The first eight features above are relatively self-explanatory. The last two, however, require some explaining. If you choose the print option provided on the first menu, you are taken to a second menu having four additional choices for printing files. These choices are:

- 1) Print a file.
- 2) Set typewriter mode.
- 3) Specify typefaces.
- 4) Send a control code to the printer.

The first option lets you suspend whatever you're doing and print a file. CTRL-S will start or stop the print. Option 2 simply prints whatever you type as you type it just as a typewriter does. The third choice allows you to send a form feed, line feed or choose any of six possible typefaces such as emphasized or compressed. This comes in handy when the application you're running will not let you embed control codes in your document. The last selection lets you send any control code to your printer. These codes are entered as a three-digit decimal number, each followed by an ESC character.

Pop-Up Anything

Naturally I saved the best for last. Somewhere in the beginning of the article, I mentioned something called *Pop-Up Anything*. Well, seeing is believing, and, believe me, it's true. This module, which takes up 20K, actually lets you make a pop-up of many of your favorite applications. It states in the manual to check your programs before using them with *Pop-Up Anything* since some cannot be popped up and others will not let you access the Popany prompt. The two I tested with were *Lotus 1-2-3* and *WordStar* and they seemed to work fine.

Pop-Up Anything is loaded by typing POPANY followed by any of several options. These options include a memory allocation, changing the invocation key, a graphics option to allow use within a program using graphics and a "risky" option if memory is in short supply. This last option saves about 6K,

however memory contents may not be as stable as you'd like. In order to obtain online help with this module you must have the *Pop-Up Anything* disk in the drive and type POPANY ?=??. None of the other modules requires the disk in a drive. Memory requirements are determined by the applications you plan to run as pop-ups.

Pop-Up Anything will also run many of the standard DOS functions. This gives you the ability to temporarily leave a program to format an extra disk if you'd like and then return to exactly where you left off. If you're like me, you'll find this extremely useful.

As mentioned earlier, using *Pop-Up Anything* in combination with the clipboard enables you to copy information from one application to another. For instance, this makes it quite easy to include spreadsheet data in a text document.

Let me give an example of using this feature. Begin by loading your word processor and a document. Next invoke *Pop-Up Anything* with ALT-F10 to receive the Popany prompt and load your word processor and a second document. Now you may work on the second or cut from it and paste to the first using the clipboard. To do this, invoke the clipboard with ALT-B and use F3 to view your document. Choose the block you want and copy it to the clipboard. Exit the word processor and receive the Popany prompt again. Type EXIT and you will be returned to your first document right where you were originally. Call up the clipboard again and reverse the process by copying from the clipboard to the document. It took longer for you to read this paragraph than it did for me to copy it from one file to another.

Conclusion

Pop-Up DeskSet has finally put my desk on the moon. It is an excellent buy and really does the job. If you have a computer on your desk, *Pop-Up DeskSet* is the thing for you. In comparison with the other desktop tools I have seen, *DeskSet* rates with the best. The one feature I miss, though, is a Hex-to-binary-to-decimal conversion routine on the calculator. I have to look out for all you programmers out there too, don't I?

(Bellsoft, Inc., 2820 Northrup Way, Bellevue, WA 98004, \$69.95)

— Larry Birkenfeld

Models III 4 4D 4P 1000 1200
Computer Aided Drafting

Save time and increase productivity

Excellent system benefit / cost ratio

Drawing sizes up to 24" x 36"

Full size overlays

Direct and auto zoom and pan

Decimal and "customary" scaling systems

Full inch and fractional grids

Cursor snap to grid or intra-grid

Variable cursor speed control

Single key functions from keyboard

Mnemonic function symbols easy to learn

Single operating mode at all times

On-line function summary, memory alert

Accuracy up to 1/100 of an inch

Digital (coordinate) input functions

Block reduce/enlarge 1% through 9999%

Block rotate 1 through 360 degrees

Block mirror / rescale

Merge, rotate, rescale drawings

Save / load drawings to / from disk

Cut out and save parts of drawings

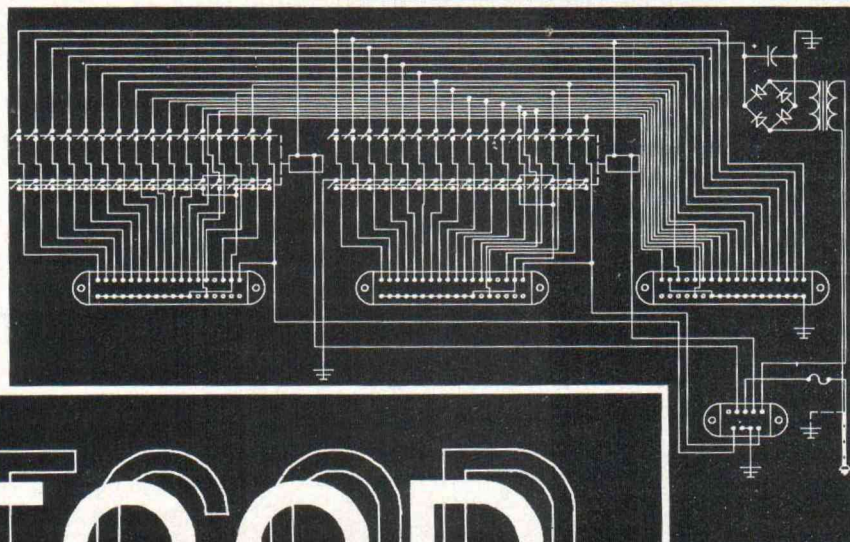
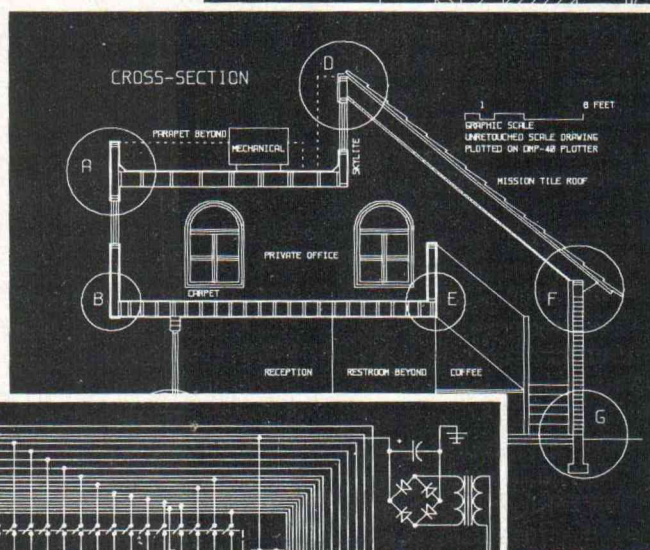
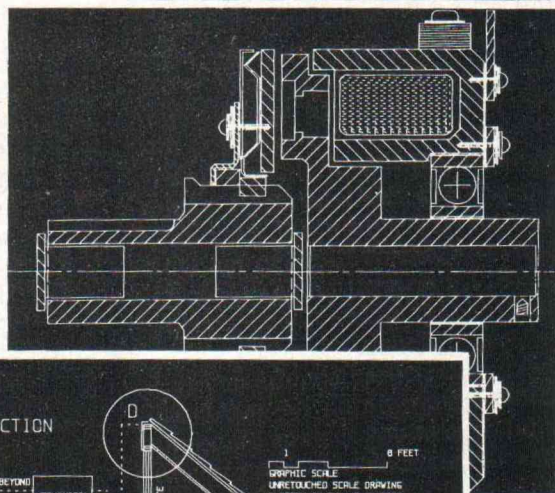
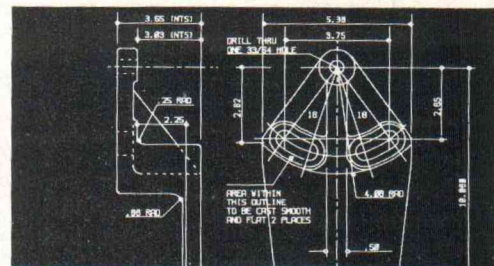
Text labels full upper / lower ASCII

Text duplication saves memory / typing

Left justified "typewriter" text option

Precision plotter output

Hardcopy reduce / enlarge 1% - 9999%



T.CAD

**Computer Aided Drafting systems
for
architects, engineers, educators,
manufacturers, builders, students ...**



MICRODEX

1212 North Sawtelle
Tucson, Arizona 85716
Phone (602) 326-3502

Offix: A File Cabinet on a Disk

Offix is an interesting program that attempts to answer one of those "why can't they . . ." kind of questions we've all been asking since the advent of the personal computer. Many people have extensive amounts of simple data to organize; people who run small businesses or large clubs often have to keep track of price lists, order information, credit information, suppliers', customers' or members' addresses and schedules. Such information piles up endlessly and must be filed, but also must be accessible. Such people need to get letters, messages or reports out, and they would benefit greatly if they could organize names, addresses and basic data, call these out of the computer easily and produce form letters and/or reports that drew on the data.

The specific question *Offix* attempts to answer is, "Why can't someone come up with a simple and inexpensive filing program that will allow me to enter, manipulate and access data in some relatively direct fashion without my having to learn a programming language?" And for the most part, *Offix*'s "user interface" will overcome the fear and trembling of all but the most terrified computerphobes. In fact, *Offix* seems, at first, like the answer to a prayer; and though the program has flaws, the first impression is substantially accurate. *Offix* promises to help the small office or large social organization do all of the above, and it largely delivers; be aware, however, that it will do little more than it promises. *Offix* touts itself as a data *manager*, not a *database*. Practically speaking, if your needs grow, or if you want a program to perform beyond simple data management, you will want a more capable program.

Installation (including printer) is straightforward and trouble-free since *Offix* comes preset for the Tandy machines. Upon booting the program — a procedure that *Offix*'s copy protection complicates annoyingly — the neophyte will be presented with a several-screen explanation of the program, which can be turned off when confidence accumulates. Following this will be a screen

that resembles a pair of three-drawer file cabinets, one of which is highlighted. One of the drawers is labeled, "Miscellaneous" and contains several *Offix* information files, including one that prescribes a simple way to set screen colors if you have a color monitor.

Offix comes with a short, soft-cover manual, a pair of duplicate, copy-protected program disks, a page of errata telling you to ignore some of the manual's instructions, and Tandy's ubiquitous software-registration form. The manual is typical Tandy documentation, i.e., not very well done, but the program offers excellent, on-screen tutorials, clear pop-up menus, and copious, context-sensitive help screens. These latter two are called by using function keys: F1 calls the simple-to-comprehend menus, and F2 calls the clearly written help screens. Any menu or help screen that comes up will relate to wherever you are in the program when you ask for help. In addition to the tutorials, there are calm-sounding and helpful error messages.

"Offix makes it easy to generate simple form letters and columnar reports"

You operate *Offix* by pressing function keys or specific (and with only one exception, logically mnemonic) alphabet keys. For instance, you type 'O' at the file cabinet screen to Open a highlighted drawer; you Rename the drawer with 'R'. You enter the word processor with 'W'. With the drawer open, you will see an alphabetical list of the first eight "folders" within the drawer. You then highlight the name of the folder you wish to (T)ake out of the drawer and then (O)pen the folder. You will next be presented with a list of the first 10 forms (in alphabetical order) within that folder, one of which you can select to scan or to change its contents. You can take out two folders at once from the same or different drawers. After you have finished with a folder, you (P)ut it back in the same or another drawer, (C)lose the drawer, and either proceed or exit. For each folder, you will design one master form that determines the format of each filled-in form in the folder. A folder will hold as many forms or documents as disk space permits. Each working disk can organize

six "drawers," each drawer can contain 100 "folders," and each folder can hold any number of forms, determined only by space available on the disk.

Forms and documents are (M)ade using the program's built-in word processor. (Note: *Offix* generates and reads ASCII files, so you can export *Offix* files or import ASCII files generated by other programs.) Though not full-featured — it lacks footnotes, super/subscript, automatic centering of text, footers, and headers, although you can assign a "page title" to each page through the print-layout screen — the *Offix* word processor is sufficient for most business correspondence. (Note: *Offix* documents are created in memory and so can be quite long; also, the program seems to take advantage of all available RAM. When I used the F7 key to see how much room I had left, a prompt informed me that on my 384K Tandy 2000 I still had 173,000 "characters available" in that particular document. That's about 100 double-spaced pages. But the complexity of long business reports would eventually drive you to a more-capable word processor.)

Nearly as simple to use as the rest of the program, the word processor will underline and boldface (although not on screen), highlight and move/copy/delete lines/blocks of text, switch from insert to type-over mode, insert/delete blank lines, and reformat the whole document (see below) or individual paragraphs. Cursor keys, PG UP, PG DN, HOME, END, DEL, and BACKSPACE all work logically, and you can use function keys to move to the beginning or end of a line. *Offix* is *not* a what-you-see-is-what-you-get word processor, although margin settings will be reflected on screen.

For printing, you set margins and such on a page-layout screen that makes the document formatting process delightfully easy. ALT-P calls up a screen that resembles typing paper. At various locations on the screen, corresponding to their positions on the final printed version, are all the various settings: margins (left, right, top, bottom), page title, page numbering, page length, spacing (up to triple spacing). You move the cursor to the appropriate area and provide the program with the numbers you want. You can modify settings during a writing or editing session via ALT-L. After formatting, F3 brings up a Print Destination Menu that gives you the choice of printing to screen (so you can see better what the final output will

be like), to disk, or to one of the defined Tandy printers.

Designing an *Offix* form is easy (a form, however, cannot be longer than *Offix*'s 22-line screen). You type 'M' at the Open Folder menu and *Offix* puts you directly into the word processor; you use this to type "labels" — data categories — followed by blank lines (which actually function as a kind of window) to create a form that is similar in appearance to a paper form. If a folder already contains a master form, you will be asked if you wish to modify it; doing so will change *all* the forms already in the folder. Interestingly, if you erase a label on a master form, any information you have previously entered for that label will stay in the form, but you won't see it. You can put the label back in at a later time and find your information on display once again. Regardless of the length of the blank line you initially enter on the master form (anywhere from 1-50 blanks), you can actually enter up to 250 characters of information on any one line (except the first, which can only be 23), then scroll along the line reading your information. You can also enter unlabeled blanks.

Typing 'S' at the Open Folder screen calls up the Search function, with which you can locate individual forms — for viewing/modifying or merging into form letters or reports — that match selected criteria. You can select all forms, request exact matches, or use asterisks and question marks as wildcard indicators. You can search for numeric matches using equal, greater-than, and less-than symbols (decimal numbers and/or alphabetic characters in the searched entries are ignored). You can also use negation: a tilde (~) tells the program to find entries that *do not* begin with a certain character.

Offix makes it easy to generate simple form letters and columnar reports, though you are limited to drawing information from within an individual folder. Though the manual is particularly confusing here, the on-screen tutorials and help will gently guide you through both of these processes. Essentially, all you need do is type into your letter the labels you wish to draw from, inform the program of search criteria, and print. It is nearly as easy as it sounds. Generating a report involves a few more steps — defining column titles and setting column widths — but is also extremely easy. In reports, *Offix* will usually sort alphabetically by the first

column, but you can instruct it otherwise just by typing an exclamation point in front of the chosen column title. *Offix* will also total numeric columns if you place a plus sign after the column title. Both form letters and columnar reports can be printed to screen so that you may judge the output before you commit hard copy.

There are several problems with *Offix*. In the words of the RSCC catalog, *Offix* "mirrors the familiar office environment." And "mirrors" is, unfortunately, exactly the right word. The program so slavishly imitates the office environment, so thoroughly insinuates the file-cabinet metaphor, that it makes it awfully simple to reproduce one of the worst faults of that system: It is far too easy, if you are careless, to "lose" a file by misfiling it, and one must then search for it exactly the same way one would search for misfiled paper — folder by folder. A revision of the program would do well to include something like a "tree" function to outline the organization of the "cabinets."

The user is several times advised to back up all work, and the warnings are intense in tone. Backing up business data is, of course, crucial, so much so that one wonders why there is no procedure for performing this routine chore from within the program. Instead, one must do backups through DOS. And in addition to backing up your *Offix* data disk, you'll need to back up the `Root.ofx` file located on your working program disk (see below about working program disks). `Root.ofx` is the file that keeps track of the organization of your cabinets and folders, i.e., which folder in which drawer in which cabinet on which disk can be found the document you want. A wipe-out of this file without a backup would be catastrophic. It's surprising that a program that is otherwise so helpful is so unhelpful here.

Re: working program disks

Offix is copy-protected; the program uses a scheme in which a garbage sector is deliberately hidden on the program disk; when booting, the program searches for this sector and either finds it or scuttles the boot. Therefore, you must boot the program with the original disk. But because *Offix* is intended to run out of the box on the 1000, 1200, 2000 and 3000, the program is provided on a 360K-formatted disk, on which there is no room for DOS. This essentially means that you will need to boot

the system with DOS, boot the program with the original or backup distribution disk, then run the program with the working disk that you can make by transferring all the files except the weird one. If you make a working disk in Tandy 2000 format, there will, of course, be room for DOS, but the program will not boot because of the lack of the garbage sector. The copy protection also causes annoyance when exiting because the program requests a copy of `Command.com`. To spare yourself this annoyance, you can copy just `Command.com` onto your working program disk. (Note: This will *not* make a bootable system disk. It will only save you the trouble of finding a system disk so you can exit.)

Miscellaneous grumblings

Once you have proceeded "down" a list of folder contents, it is sometimes difficult to go back "up." Frequently, you will have to use the HOME key to return to the top of the list before you can go back down. This can be extremely frustrating if you are near the end of a long list and need to go back up just a few entries and find you must go to the top just to get back down to the bottom.

When I reformatted a letter with the cursor at the top, the program reformatted the address block so that it all came out on one line.

When I booted *Offix* after a session using *Videotex Plus*, that program's alternate-function-key settings remained in memory, so that when I accidentally typed ALT F1, my MCI Mail number popped into my document. In fact, all 12 of my *Videotex Plus* presets were still in memory. It appears that *Offix* doesn't completely clear memory when it boots. Since this works so well, *Offix* upgrades should definitely include the ability to use ALT/function-key settings to record frequently used phrases or numbers. This would enhance the program's usefulness.

A section of the manual informs you that you can write BASIC routines that will act upon the ASCII files that *Offix* generates. The manual then tells you that how much you can get out of these routines depends on your ability as a programmer. But it seems reasonable to assume that a user who learned to write BASIC routines complex enough to go beyond *Offix*'s limits would not any longer need *Offix*.

Within the list of printer definitions,

although the DWII is listed, the DWIIB is not, and in fact a DWIIB would not operate on the DWII setting; it did operate flawlessly from generic DWP 10-pitch and DWP 12-pitch settings. And the errata page reports that the printer table for DMP Standard 10 and 12 are reversed.)

Most of what *Offix* does, it does relatively painlessly — so painlessly, in fact, that after a while, user-friendliness becomes user-obsequiousness. All that friendliness begins to get in the way. Also, you will occasionally get caught up in a surprising carom across the border into deepest user-hostility, as when you discover the extremely arcane and confusing section of the manual that describes how to enter numeric-comparison data for searches. But for the most part, *Offix*'s "user interface" will overcome the fear and trembling of all but the most terrified computer-phobes.

Offix, as it comes out of the box, is a capable, if limited, filing program. What *Offix* does do, it does nicely. It is easy to learn and easy to remember. It will organize and track a virtually unlimited amount of simple data and

create simple form letters, mailing lists and reports with relative ease. Its word processor, though a bit clunky, is easy to use and capable of generating simple business documents. As your needs grow in complexity, you will quickly outgrow *Offix*. But it would not be fair to criticize *Offix* for what it is not, i.e., a full-scale database for the generation

of complex statistical analyses. Judging *Offix* in terms of what it is — a capable data manager — earns the program generally high marks.

(Tandy Corp., One Tandy Center, Fort Worth, TX 76102, \$99.95)

— Gary Karasik

Software

1000

PowerText: Word Processing Enhancement

PowerText is aptly named, it adds powerful features and options to your word processing capabilities. Word processors offer a variety of features for formatting text which varies from one word processor to another. Generally, the more costly word processors offer more features than the less expensive

ones. Some are more difficult to use than others, or I could say, less user friendly. Usually the more features that are offered, the more difficult it is to learn, or at least there is more to learn.

Regardless of how many features are offered by the word processor, it doesn't take us long to wish for more. That's where *PowerText* comes in; it adds many features to even the simplest of word processors. It takes simple commands to produce what would otherwise be a complex typing job. *PowerText* is ahead of the field in providing powerful formatting features for word processing outputs.

PowerText offers options which include left- and right-justified text, automatic title pages and tables of contents for papers, font variations and

MSDOS UTILITIES Tandy 1000, 1200, 2000, 3000 IBM PC/XT/AT

(No Technical Knowledge Required! Easy to Use!)

***** RAMDISK *****

\$49

Use spare memory to simulate one or more superfast disks. Any size. Greatly speed up existing applications. Appears to programs as a disk, but gives instant access. Highly efficient, compact, and thoroughly tested. Includes program to install and change RAMDISK size - no need to use complicated edit programs.

***** FULLBACK *****

\$88

(A Perfect File Backup System!)

Finally, an easy to use backup program that keeps exact images of your files on backup floppies, cartridges, or hard disk. Automatically backup one, several, or ALL subdirectories. Backup modified only, or ALL files. Keeps perfectly organized backups - backup structure is identical to original. Supports backup by date, multiple backup copies, large files (up to 32MB). Far superior to DOS BACKUP, easier to use, and much more reliable. Absolutely a MUST for hard disk users.

***** USER TOOLS *****

\$59

(Save time, improve performance, increase flexibility)

KB -- Keyboard aid, generate text with single key.
DI -- 5 column, sorted, fast directory display.
FINDFILE -- Search all directories for file.
DTREE -- Beautiful subdirectory display (Names, size)
CHMOD -- Set file attributes. (Readonly, Hidden,...)
HEXDUMP -- Display file in hex and character format.
TI -- Report elapsed time for any DOS command.

More ALPS Tools Available. Call or Write for ALPS catalog. Customer Support!!! 9am to 5pm weekdays. ALPS develops and supports all products in-house, and has 5 years of business experience with microcomputers.

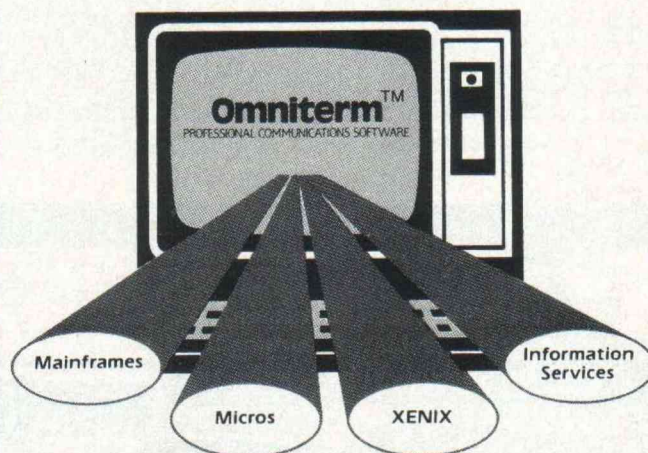


ALPS

1502 County Road 25
Woodland Park, Colorado 80863
303-687-1442



Make Great Connections!



- Transfer files directly to/from disk
- Use XMODEM for error correction
- VT100 terminal emulation
- Autodial and logon with any autodial modem

Now available in Radio Shack Stores! Ask for

Omniterm 2
Cat.# 25-1160

\$129.95

IBM PC/XT/AT

Tandy 1000/1200HD/2000

Omniterm Plus

EOS Cat.# 900-0123

\$95.00

TRS-80 Model 4/4P

Omniterm is a product of Lindbergh Systems, Inc.
95 Nagog Hill Road, Acton, Ma 01720 (617) 263-5049

automatically generated outlines with proper numbering and indentation. It will also give you multiple columns of text, as in newspapers, which can be synchronized from one column to the next if you so desire, complex headers and footers, footnote features, screen-play script with proper formats, complex pitch, line spacing and margin choices, calendars/schedules, form letters and more.

PowerText uses the text file created by your word processor with embedded *PowerText* commands to produce complex documents. With the large variety of software available today, I can't say that *PowerText* has features that aren't found in any other word processor. But, it is safe to say that most word processors, especially the relatively simple and less expensive word processors that most of us use at home, don't offer the wide combination and variety of features found in *PowerText*. You would be hard pressed to find even an expensive word processor for business use which offers the features found in *PowerText*. And get this: The price is only \$49.95 plus \$5 shipping.

In fact, *PowerText* is designed to import files from a variety of word

processors commonly found in offices for business use. More specifically, *PowerText* is setup to take text files from *pfs:write*, *Writing Assistant*, *WordStar*, *Volkswriter*, *Multimate*, *Wordvision* and any ASCII editor.

PowerText is designed to handle the above six specific word processors directly, or any other word processor that uses ASCII text files. That should cover the ball park pretty well. Almost any simple text editor or word processor can be used to prepare an ASCII text file for use by *PowerText*.

PowerText comes with two disks and nearly 300 pages of documentation. It runs on IBM PC/XT/AT computers and the compatibles with 192K and either two disk drives or hard disk (not copy protected). My Tandy 1000 with 256K worked fine. It's simple enough to use if you follow the directions. I tried to jump right in after only half glancing at the documentation and didn't produce much. When I actually took a good look at the documentation and the samples provided, everything suddenly became much simpler.

PowerText comes with 16 sample files to show off the large variety of features available and how to use them.

Considering that it comes with the samples, quality documentation and on-screen prompts/instructions, it shouldn't be very difficult to get started. Part of the documentation steps you through a procedure to list the various sample text files straight to the printer with the DOS copy command `COPY SAMP2 LPT1:` and then to print the same file using *PowerText*. The difference in the two printouts is generally very dynamic. It is really amazing how little needs to be placed into the text file in the way of embedded commands to produce such nice outputs.

The typical routine of building a working disk is the first step in getting started. This is well covered in the documentation. Part of this is selecting a printer driver from among the variety of options provided. These drivers can be modified, or a new driver can be built by using the Install feature and following the documentation instructions that go along with the menu prompts.

Building a printer driver requires the entry of decimal ASCII codes used by your printer. For example, at the prompt that asks for codes to turn on the superscript feature of your printer, I enter "27 14," which represents the

Only \$24.95

COCO-UTIL

"The Connecting Link"

CoCo-Util is a valuable utility program that allows you to transfer Tandy Color Computer disc files to your MS-DOS machine. You may also transfer MS-DOS files to a Color Computer disc. CoCo-Util will save you countless hours of retyping... a great new utility.

SALE

64K Upgrades Set of 9 chips Great Bargain

Only \$14.95

PC-XT II

We now offer IBM PC-XT compatible computer systems. Don't settle for a Tandy 1000 or other standard PC until you check our systems and prices.

Starting at \$865.

MS-DOS ADVENTURES

Our popular Hi-Res Color graphic Adventures... Callisto Island, Black Sanctum, Shenanigans, Sea Search, Trekboer are now available for MS-DOS computers to challenge and entertain you.



Only \$29.95 per disc

SHIPPING: All orders under \$100 please add \$2 regular, \$5 air. All orders over \$100 please add 3% regular, 8% air. California residents please add 6% sales tax. Orders outside the continental U.S. check with us for shipping amount; please remit U.S. funds. Software authors—contact us for exciting program marketing details. We accept MasterCard and VISA. Distributed in Canada by Kelly Software.

PRINTERS

New Star Gemini SG-10 Model printer. 120 cps w/true descenders, proportional spacing. New model for 1985.



Our Great Price, Only \$249.95

QUALITY VIDEO MONITORS

We carry both monochrome and color monitors at the best prices—several brands available. Call for more details.

Priced from \$109.95

TILT SWIVEL BASE

Now you can adjust your monitor for best viewing angle. Left, right, up, down. Use with any monitor. Compare with units that sell for much more. Our great price—**Only \$14.95**

DISC STORAGE CASE

Attractive, heavy duty, smoked acrylic case. Holds and protects up to 75 5¼" discs with six moveable, indexed dividers.

SUPER BARGAIN ONLY \$9.95
storage box along with ten discs for only \$21.95.



10 5¼" discs in attractive styrene box—**Only \$12.95**



Mark Data Products

Department I

24001 ALICIA PKWY., NO. 207 MISSION VIEJO, CA 92691 (714) 768-1551

FREE—Send for Our Latest Product Flyer!

commands for my particular printer. These commands are normally listed in the back of the printer manual. Building your own custom printer driver takes you through quite a series of such prompts; and, based on my experience, this may take a few tries to get it right. Fortunately for those who feel building a printer driver is too technical, there is a default driver. In addition, one of the specific drivers provided with the software should come fairly close to the features of most printers. If all else fails, a call to the software developer should solve any problems here.

In addition to the text file with embedded *PowerText* commands, a style or format file to define the type of document you are producing is necessary. Building a style file could take some effort if you were to attempt it from scratch using just the documentation as a guideline. However, this problem is readily solved by the software, which comes with a variety of style files used to produce the sample documents already mentioned. It's a relatively simple job to select one of the style files provided and to make a few modifications to it for your specific application. In many cases, no modification would even be necessary. The style file is changed, listed or created by use of your word processor. The first embedded command in your text file refers to the particular style file you want to use.

Most of the embedded commands start with a slash (/) as the first non-blank character of the line. In this way, the slash can be used for normal purposes, such as in the date 1/15/86, without being confused with an embedded command. A few of the embedded commands use the caret (^) to invoke a *PowerText* feature. Here are a few examples of embedded commands:

/adj	sets the text to right justification
/adj [5]	sets the text to right justification to within five characters of the right margin
/block [n]	reserves a block of empty space in your document with the 'n' representing the size of the block
^b text ^	sets the text within the "b" and caret commands to bold printing
/box cen	places the paragraph of text in a centered box
/col	a powerful command for

	producing up to 14 columns with specifications for the column width, style, etc.
^f text	footnotes of up to 87 lines to be placed at the end of the page or end of the document with whatever type style you desire
/inc file	allows other files to be included in a document
/sal	means salutation, one of a number of keywords for constructing headers, footers, closings and cover pages
/news	used to divide the page into newspaper like columns for journals, newsletters, etc.
/out	used to automatically build an outline with many options
/tp	forms a paragraph with a centered title for plays and screenplays

/2 Breakthroughs
 /3 Hardware
 /3 Software
 /3 Here is when more than one line is formed into a paragraph.

The printed output from *PowerText*:

```

Sample Outline
(I) The Historical View
  (a) Prior to 1900

    Here is a boxed paragraph
    formed per current margins

  (b) Post 1900
    (1) Pre-Industrial Rev.
    (2) Post-Industrial Rev.
(II) The Technical View
  (a) Technology Evolution
  (b) Breakthroughs
    (1) Hardware
    (2) Software
    (3) Here is when more than one
        line
        is formed into a para-
        graph.
```

A quick review of just some of the samples that come with *PowerText* will provide some insight into its wide range of features.

Sample 1 uses embedded commands to turn a one-page text file into a four-page finished report with a title page, a table of contents and two pages of text with centered and unlined headings.

Sample 3 expands on Sample 1 with two additional embedded commands that add centered blocks of quoted text in a different type style and add footnotes to each page.

Sample 5 uses the report style file and the outline command to build a well-formed outline, including a boxed paragraph of special information. Here is a miniature example giving the text file input into *PowerText*:

```

<report
<cen ^b Sample Outline^
<out [ra(n[
/1 The Historical View
/2 Prior to 1900
<box Here is a boxed paragraph
    formed
    per current margins
/2 Post 1900
/3 Pre-Industrial Rev.
/3 Post-Industrial Rev.
/1 The Technical View
/2 Technology Evolution
```

Sample 6 varies the outline by using "bullets" or asterisks (*) instead of the standard numbering and lettering of the outline items.

Sample 7 takes a simple text file which has some paragraphs starting with "/1" and others starting with "/2" to build a document using the column command. The end result is a document with matched columns of text intermixed with full-width text. Some columns are divided by a vertical line while others are not. Selected areas of the document are boxed. By matched columns, I refer to information in one column which starts on the same line as matching information in the next column, regardless of the fact that the length of the entries varies from one column to the next. I might point out that none of the alignment was handled by the typist. The alignment and arranging of the text was all handled by *PowerText*.

Sample 8 takes a page of normal text with six embedded commands to produce a one-page newsletter type of document. The title is centered and underlined, the opening paragraph is the full width of the page, and the rest is divided into two columns. The two columns include a subject title centered within one column and several paragraphs of text, right-justified within the

columns. I have had some experience preparing a similar newsletter type of format with a typewriter. Believe me, this feature will be appreciated by anyone preparing a newsletter.

Sample 9 produces a letter with automatic page breaking, centered heading information on the first page, continuation headings on subsequent pages, a footnote and centered closing. By referring to automatic page breaking, I don't just mean so many blank lines left at the bottom of each page. *PowerText* will fit the letter onto one page if it can, even if the footer space requirement has to be violated. If the letter will not all fit on one page, it will arrange the letter so that it looks nice. There will not be one line moved to the last page all by itself.

Sample 11 uses the schedule command to build a calendar style schedule. There is a column for each day of the week with boxed-in areas under the column headings for the various hours of the workday, including scheduled appointments.

Sample 13 is for merging other files into a book with each separate file becoming a chapter.

Sample 15 uses the play style. "\g" is used to designate the beginning of the girl's statements and "\b" is used to designate the beginning of the boy's statements to produce a very nicely laid out document with script.

Sample 16 uses a name and address file (Sample 17) to produce form letters with the name, address, etc., selected from the the address file. Embedded commands in the letter cause the first name of the related record in the name and address file to be positioned at various places within the text of the letter.

In summary, *PowerText* offers some very powerful features at a very low cost. It enhances even the simplest of word processors with features that would normally be available in software that is priced out of reach for most home computer use. I recommend *PowerText* for students, for anyone writing books, plays or other large documents, for newsletter editors, for secretaries who do more than an occasional letter, for anyone doing quantities of form letters and for anyone interested in adding power to their word processing capabilities.

(Beaman Porter, Inc., 417 Halstead Ave., Harrison, NY 10528, (914)835-3010, \$49.95 plus \$5 shipping, \$10 outside U.S.)

— Robert Jensen

Software

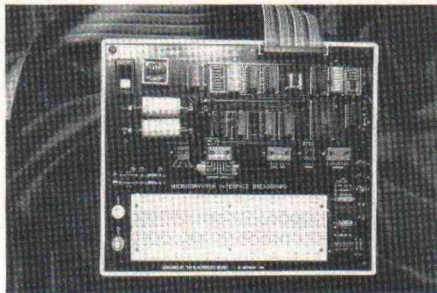
1000/1200/2000

Diagram Master Eliminates the Tedium of Drawing Charts

DecisionResources, Inc., of Westport, Connecticut, has released a new program, *Diagram Master*, for creating simple drawings, organization charts and gantt schedules. If your job involves creating such materials, then you might want to consider using *Diagram Master*. It is another in their "Master" series that includes *Chart Master* and *Sign Master*.

Diagram Master combines some unique features that make it an interest-

**Our new Interface Card lets you
connect your MS-DOS computer to the
outside world to control and monitor
devices and instruments**



The BG-Board Interface System provides buffered access to the control, data, and address buses of the computer. Specially designed CableCards orient the signals to and from the BG-Board and many popular computers. The new BG-119 CableCard brings the MS-DOS machines into the fold with the previously available CableCards for the Model III/II, Apple II, Commodore 64, and VIC-20. Our new catalog describes the system and more than 100 books on interfacing, electronics, and personal progress plus some scientific and utility software that can simplify data analysis and computer performance. Write or call for it and the special sale now in progress. 703-651-3153



Group Technology, Ltd.

P.O. BOX 87 • CHECK, VIRGINIA 24072

Trademarks - MSDOS, Microsoft. Apple II, Apple Computer, Inc. Commodore 64 and VIC-20, Commodore Business Machines, Inc.

EXPAND THE TANDY 2000®
TO

896K

OF DOS ADDRESSABLE RAM

Includes RAMdisk Software at No Extra Cost ...
Add 640K of External RAM Memory to TANDY 2000's®
256K of Internal RAM Memory for a total of 896K of DOS Ad-
dressable RAM.

★ ★ New Products Available Soon ★ ★

(Send for catalog)

cost:

\$495.00 for a 640K bytes of External
RAM Memory Board (Includes RAMdisk
Software)

Call about our Trade-in Exchange option
if you already use Tandy's 128K External
Memory Board.

for ordering (509) 627-5291
and information call: Hours 9:00-5:00 PST

terms:

We accept Check, MO, Visa, Mastercard
Add 7.3% sales tax in Wash.
Add \$3-shipping/handling, \$5 foreign.

TANDY 2000® IS A TRADEMARK OF TANDY CORPORATION
RAMdisk is a Random-Access-Memory based disk

ENVISION DESIGNS

1909 ORCHARD WAY • RICHLAND, WASHINGTON 99352

ing, but not always impressive, software package. And while it may meet your needs, it may not meet them on time, as I found it to be slow and cumbersome to use.

Diagram Master, geared toward business managers and organizations, provides you with the power to create and edit business diagrams. You may use it to create flow charts for software, processes and organizations. Also you may create organization charts to graphically show "who is reporting to whom." Plus, *Diagram Master* gives you the ability to set up gantt charts for illustrating schedules of projects and activities.

Diagram Master comes with diskettes, a thorough loose leaf manual and a free subscription to their newsletter.

"Using *Diagram Master* is very simple and almost entirely menu driven."

ter. In addition, DecisionResources provides free technical support on their technical hotline at (203) 222-1974. I had occasion to use this hotline and they were quick, courteous and helpful. Let's take a look at what is provided on the diskettes and in the manual.

Of the five diskettes provided, only one is copy protected and contains the main program of *Diagram Master*. If you own a hard disk, though, you may install the copy protected software on your hard drive along with the rest of the support files on the other diskettes. The process is simple, well defined in the manual and includes a way to copy protect your hard drive copy, so that inserting the main program disk is eliminated. Otherwise the original main program disk must be in Drive A in order for the program to run.

The other four diskettes contain support files for *Diagram Master* including a diskette of already completed drawings and charts called the Library Disk. The other three diskettes contain Drawing Board, Gantt Chart, and Organization Chart files, including samples on each disk respectively. During operation of *Diagram Master*, you are required to insert these disks at various prompts in order to perform certain tasks. I highly recommend a

hard drive for efficient use of *Diagram Master*.

The minimum system requirements for *Diagram Master* are a PC or PC compatible, two floppy drives or hard disk, a minimum memory configuration of 320K (384K for printers and Polaroid Palette) and a video display (color or non-IBM monochrome) with a graphics adapter.

The hardware that *Diagram Master* supports is one of the most impressive features of this software. It not only supports most of the best selling printers on the market, but, also supports many of the best selling plotters and the Polaroid Palette. This allows those users with plotters to get impressive hardcopy and to get slides for large professional presentations or 3.25 by 4.25-inch color prints using the Palette. *Diagram Master* supports a light pen as an alternate input device. I don't own one and was unable to check this feature; however, the manual covers the set-up and calibration of your light pen.

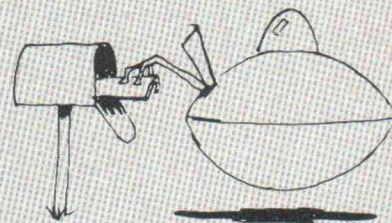
Using *Diagram Master* is very simple and almost entirely menu driven. When you start up, you are prompted to answer a series of questions concerning the system and peripherals you will be using. You may easily change these settings later if your system changes, and once you finish this installation you are greeted with the opening menu of *Diagram Master*.

The main section of *Diagram Master* is called Drawing Board, and provides you with the tools to create simple drawings out of predefined shapes, lines, various type styles and hatch styles. The process is almost entirely menu driven and requires that you only answer prompts to achieve the tasks at hand. In addition, you may edit and add to organization and gantt charts using the Drawing Board section of *Diagram Master*.

The two other main features of *Diagram Master* are Organization Charts and Gantt Charts. Each of these sections also operates using menus and simple prompts, except at this point data is input concerning your specific chart.

For organization charts, you enter data on how many levels, boxes, layout style and text type you will be using and the specific data to appear in the boxes of your chart. You may create new charts or edit existing charts. You can also globally change the text size on your chart as well as print, plot, save and retrieve charts from a diskette. In

PCM



Back Issue Availability

Back copies of many issues of PCM are still available.

All back issues sell for the single issue cover price. In addition, there is a \$2 charge for the first issue plus 50 cents for each additional issue shipped in the U.S. When possible, issues are shipped UPS. The postage cost in Canada and Mexico is \$3 for the first issue and \$1 for each additional issue.

VISA, MasterCard and American Express accepted. Kentucky residents please add 5 percent state sales tax. We do not bill and no COD orders accepted.

We suggest you order back issues you want now while supplies last.

To order, just fill out the form on the next page and mail it with your payment to:

PCM
The Falsoft Building
Prospect, KY 40059

☐ Please send me the following back issues:

(Payment must accompany back issue orders.)

VOLUME 1			
NO. MO.	YR.		PRICE
1 JULY	'83	PREMIER ISSUE	\$3.00 <input type="checkbox"/>
2 AUG.	'83		\$3.00 <input type="checkbox"/>
3 SEPT.	'83		\$3.00 <input type="checkbox"/>
4 OCT.	'83		\$3.00 <input type="checkbox"/>
5 NOV.	'83		\$3.00 <input type="checkbox"/>
6 DEC.	'83		\$3.00 <input type="checkbox"/>
7 JAN.	'84		\$3.00 <input type="checkbox"/>
8 FEB.	'84		\$3.00 <input type="checkbox"/>
9 MAR.	'84		\$3.00 <input type="checkbox"/>
10 APR.	'84	LIMITED	\$3.00 <input type="checkbox"/>
11 MAY	'84		\$3.00 <input type="checkbox"/>
12 JUNE	'84		\$3.00 <input type="checkbox"/>
VOLUME 2			
2 AUG.	'84		\$3.00 <input type="checkbox"/>
3 SEPT.	'84		\$3.00 <input type="checkbox"/>
4 OCT.	'84		\$3.00 <input type="checkbox"/>
5 NOV.	'84		\$3.00 <input type="checkbox"/>
6 DEC.	'84		\$3.00 <input type="checkbox"/>
7 JAN.	'85		\$3.00 <input type="checkbox"/>
8 FEB.	'85		\$3.00 <input type="checkbox"/>
9 MAR.	'85		\$3.00 <input type="checkbox"/>
10 APR.	'85		\$3.00 <input type="checkbox"/>
11 MAY	'85		\$3.00 <input type="checkbox"/>
12 JUNE	'85		\$3.00 <input type="checkbox"/>
VOLUME 3			
1 JULY	'85		\$3.00 <input type="checkbox"/>
2 AUG.	'85		\$3.00 <input type="checkbox"/>
4 OCT.	'85		\$3.00 <input type="checkbox"/>
5 NOV.	'85		\$3.00 <input type="checkbox"/>
6 DEC.	'85		\$3.00 <input type="checkbox"/>
7 JAN.	'86		\$3.00 <input type="checkbox"/>
8 FEB.	'86		\$3.00 <input type="checkbox"/>

TOTAL _____

SHIPPING & HANDLING _____

KY RESIDENTS ADD 5% _____

TOTAL AMOUNT
ENCLOSED _____

NAME _____	STATE _____	ZIP _____
ADDRESS _____		
CITY _____		
CARD # _____		
EXPIRATION DATE _____		
SIGNATURE _____		

PAYMENT ENCLOSED

☐  

☐ 

this section of the manual there are examples of the different types of charts that can be created and simple lessons on creating a chart.

The third major section of *Diagram Master*, Gantt Charts, provides the menus and prompts for creating and editing these documents. What is a gantt chart? It is a time based diagram that shows the start and end points of different tasks and activities. These charts are used to keep track of projects, create schedules and plan time tables.

To create a gantt chart, you input data covering the time divisions to be used, activities, bar types, patterns, color and the format; layered or segmented. Once you've finished your chart, you may elect to view it on full screen, print/plot it or send it to the palette. In addition, you may edit the chart in the drawing board section.

An additional feature of *Diagram Master* allows you to change text fonts and sizes for your charts. There are eight fonts provided in 16 different sizes (points) from which to choose. The styles provided are Standard, Bold Standard, Roman, Bold Roman, Script, Swiss, Bold Swiss and Symbols. The Symbols font is actually not standard characters but are various pre-drawn graphics that can be inserted in your charts and drawings with a single keystroke.

A plastic-laminated keyboard guide, that comes with *Diagram Master*, gives you reference to all the available symbols and their representative keys. The symbols include some of the international symbols found on signs, along with symbols of an airplane, truck, computer, TV, happy face, sad face, clock, telephone, bell, christmas tree, American flag and various charting symbols like arrows, circles, boxes and math symbols.

In addition to the symbol font, you may use any existing drawing from your library disk as a symbol in another drawing. This allows you to create your own drawings as symbols and then place them in other drawings repeatedly. This makes *Diagram Master* a very powerful symbol editor, too.

Is *Diagram Master* for you? Will it meet your needs? Look very carefully at this package before you buy. I found it to be full of features that can be useful, but I also found some problems that may prevent *Diagram Master* from being the end-all software for your charting needs.

While *Diagram Master* supports

many output devices, the method used to create your drawings and charts is less than intuitive, and the draw and redraw times are very slow. And while the redraw times may be faster on machines with faster clock speeds, I don't see any way around the cumbersome drawing procedures and multi-keystroke menu entries.

When making a menu selection, you have the option of selecting the item two different ways. *Diagram Master* highlights menu selections with a moving highlight bar. You may move this bar using the arrow keys and then hit ENTER to make that selection. This works well, but requires many keystrokes to get from the top entry to the bottom entry. In order to avoid this, DecisionResources has placed a number beside each menu item. By selecting any number that corresponds with the selection you wish to make, that selection is then highlighted. This saves keystrokes, but still requires that you press ENTER to finish making that selection. It would be much faster if the authors had allowed you to make a single keystroke of that number to get immediate action on your selection.

I must point out that while the above is somewhat inconvenient, *Diagram Master* does have a real time-saving feature built into the menu selection process. As you are greeted with each subsequent menu, the most logical next selection is already highlighted. This speeds selection making and also helps you find your way through the many menus without referring to the manual. For example, after creating a drawing you will return to the main menu with the save-feature highlighted. This is a very nice feature of *Diagram Master* and speeds the process considerably.

My major criticism of *Diagram Master* involves the draw and redraw time. To load and then draw an existing diagram or chart is very time consuming. And, the more complicated the drawing, the longer it takes to see results. This is due to *Diagram Master* handling the drawing files by using vector graphics and re-creating the drawing based on data rather than on pixel settings. So, if you have lots of drawings with which to work, this can be a real detriment to getting results quickly. By the way, often vector graphic output speed can be increased by equipping your system with an 8087 numeric co-processor. This is not the case with *Diagram Master*.

One final major criticism I have with

Diagram Master involves the method with which you draw on screen. You are given a work space in which to work along with menus on the side to help make your selection. While this is convenient, it requires that you draw only in high resolution mode without color. You must keep track of your colors mentally. To see your drawing in full color, you must go to the preview/plot menu and preview your drawing on screen. This, in turn, involves once again the slow redraw time.

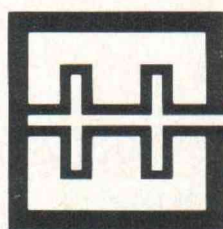
Other minor problems you should know about involve the monitor color selection, which is limited to pre-defined palettes of which you may only change the background color. In addition, you may only have six reporting levels on an organization chart. If your organization has more levels, you will need to create more charts. Limitations for creating gantt charts are less severe. You may have up to 25 activities, with a maximum of 24 major time divisions and 104 minor time divisions.

In summary, *Diagram Master* packs some very useful features into one package and may provide you with the tools you need to create drawings and charts for your company or organization. The hardware support is impressive. The manual is thorough and has many examples and lessons to help you get a grasp of all of *Diagram Master's* features. In addition, *Diagram Master* can be used in tandem with Decision-Resources other packages, *Chart-Master* and *Sign-Master*.

If you aren't in a hurry and don't mind the restrictions I've addressed, then give *Diagram Master* your consideration and see if it will meet your needs. Creating charts can be a tedious task, but with *Diagram Master* much of the tedium is eliminated.

(DecisionResources, 25 Sylvan Road South, Westport, CT 06880, (203) 222-1974, \$345)

— Bobby Ballard



HOWARD

PAYROL/BAS™

BY
BERNIE LITTON

- Professional looking
- Eliminates math errors
- No tables to look up
- Saves valuable time
- Written in non protected BASIC
- Custom code for each state
- Handles up to 500 employees, & deductions, 4 pay periods
- Automatically calculates FED FICA. No tables to enter
- Automatically calculates 5 individual deductions either straight or percentage & misc. ded
- Prints the check
- Stores all of the above data to print 5 reports including check summary, employee listing, ledger summary, Federal withholding deposits & totals
- Handles weekly, bi-weekly, semi-monthly and monthly
- Handles hourly, salary, vacation, overtime, doubletime, sicktime, bonus, and commission
- 40 page manual

Automatically Calculate State Income Tax for 49.95

• AL 49.95	• FL 0.00	• LA 49.95	• NE 49.95	• OK 49.95	• UT 49.95
• AK 0.00	• GA 49.95	• ME 49.95	• NV 0.00	• OR 49.95	• VT 49.95
• AZ 49.95	• HI 49.95	• MD 49.95	• NH 0.00	• PA 49.95	• VA 49.95
• AR 49.95	• ID 49.95	• MA 49.95	• NJ 49.95	• PR 49.95	• WA 0.00
• CA 49.95	• IL 49.95	• MI 49.95	• NM 49.95	• RI 49.95	• WV 49.95
• CO 49.95	• IN 49.95	• MN 49.95	• NY 49.95	• SC 49.95	• WI 49.95
• CT 0.00	• IA 49.95	• MS 49.95	• NC 49.95	• SD 0.00	• WY 0.00
• DE 49.95	• KS 49.95	• MO 49.95	• ND 49.95	• TN 0.00	• NYC 49.95
• DC 49.95	• KY 49.95	• MT 49.95	• OH 49.95	• TX 0.00	• VI 49.95

PAYROL/BAS™ ONLY

\$149.95

Plus \$2 Shipping

Minimum system requirements
MS-DOS

2 double sided drives
for the TANDY
1000, 1200, or 2000

WITH PAYROL/BAS™ YOU'LL ALSO WANT

941 PROGRAM
gives individual summaries
and totals of check information
to prepare 941 and state
unemployment forms

\$29.95

W-2 PROGRAM
prints out year end W-2's

\$29.95

RETURN TO HOWARD MEDICAL COMPUTERS

Box 2, Chicago, IL 60690

quantity	description	cost	extension
	PAYROL/BAS™	149.95	
	state taxes		
	941 program	29.95	
	W-2 program	29.95	
	shipping		\$2.00

include name, address, city, state and zip card #
expiration date and signature on a separate sheet
or call 312-278-1440

COD (add \$1.90)

TOTAL

312-278-1440

Software

2000

Gramarcy Enlivens Proficiency

The ultimate learning tool is one that is fun to play, and what more fulfilling benefit to receive from playing a game than to learn. *Gramarcy* is just such. It

is a computer program that plays the "word game" that appears in many daily newspapers.

Gramarcy, while being incredibly easy to play, is guaranteed to provide hours of challenging excitement. The main menu displays the three options provided: Play the game, analyze a word or modify the data bank.

After reading a simple set of rules, the computer will then ask the length of word you wish to play, ranging from

seven to 11 letters. When you are ready to start, a word will be displayed and a time limit set. The player then lists as many words with more than four letters that can be made from the word the computer chose. For example, from the word "program," nine other words can be formed: armor, gram, mora, parr, prom, ramp, roam, roar and romp.

Because, according to the documentation, "the rationale for playing the game is to become more proficient in

the spelling and use of the English language," and for those of us who do not know what a mora or parr is, it is highly advised to keep a dictionary handy.

Each player's score is composed of two parts. There are points awarded for each word correctly formed and points are subtracted for incorrect words. Bonus points may also be earned by completing the game before the computer set time limit. To help you keep track of your progress the computer will keep track of 10 high scores for 15 players.

The second selection from the main menu analyzes a selected word from four to 11 letters long. It is recommended to run *Gramarcy* on a hard disk to increase the speed, but even on a two disk system it is fast. To analyze a four-letter word it takes two minutes; an 11-letter word, five minutes. While it takes slightly longer for the computer to analyze a word while "playing the game," even then it is quick. Whatever method is used to analyze the words, apparently it is good. The programmer has applied for a patent for it.

For the professional "word game-*Gramarcy*" player, the third option allows the player to modify the data bank of words. Words from four to 10 characters long may be inserted or deleted. This allows the perfectionist to modify the data bank to his own definitive source.

Two words of warning. It is necessary to have a printer on at all times. Most output is directed to the printer. When it is not on, you'll receive no polite warnings, and the program simply exits to the operating system. Technically, it is against the official rules to form a word itself from itself. However, this over ambitious word finder does list the word itself. In our example it would have also listed the word program.

A handy 11-page manual accompanies the program. It explains the different functions of the program and the rules it plays by. It is a concise, easy-to-understand aid.

As with other Cheapware products, the price is more than right and the program works very well. So if you can find 15 words in the trademark "gramarcy" then you are ready to meet this computer (72 words) challenge (41 words).

(Cheapware, 4038 N. Ninth Street, St. Louis, MO 63147, \$35.)

— Melanie Bunger

The Model 100/200 Disk Drive is Dependable and Tough

If the Model 100 or Tandy 200 needed one thing to make them ideal lap computers, it was a portable disk drive. Now they have one.

In fact, it is hard to say enough for the new Tandy Portable Disk Drive. It is compact, well built and light, and can just fit into a briefcase with the Model 100 and a small tape recorder. It uses micro floppy disks, which are so superior to the old style floppies that any computer that doesn't use them should be considered outmoded. It only gets 100K on a disk, while micro drives for larger computers pack 500K per disk, but 100K is plenty for the tiny Model 100/200. In fact, it is amazing how much fits on a disk.

The drive is both fast and easy to use. It works at 9600 Baud, so that 10 minutes of taped data goes onto a disk in a few seconds. The floppy system has eight functions corresponding to the eight Model 100/200 function keys. These functions are a listing of all files on the disk with their byte counts (and a byte count of remaining space on the disk), a renaming function that lets you switch the name on a file on disk, plus the usual file load, kill, and save and disk backup, and format. F8, as usual, brings you back to the menu.

The drive runs on four Double A batteries. One of its few drawbacks is that it will not function on rechargeables. The directions say it will run only two hours on a set of batteries. However, the drive operates so quickly that if you are careful about only turning it on when you need to use it, you can spread those two hours out over several weeks. It will also run on the same model AC converter that your Model 100/200 uses.

To use the disk drive, you must first load a 3,359 byte machine language program, which is saved both in high and low memory, creating a 7,000 byte overhead. However, once the program

is loaded into high memory you can delete the low memory copy, saving half the overhead but removing the program from your menu. To access the program you then need the following one-line BASIC program: "10 CALL 59400" for Model 100s or "10 CALL 57500" for 200s. Save it as FLOPPY.BA. If you need to use some other machine language program that uses the same high memory space as FLOPPY.CO, you can resave it into low memory with the command in BASIC: SAVEM"FLOPPY.CO", 59400, 62752, 59400. If you use a Model 200, the numbers are different.

Even 3,500 bytes overhead seems like a lot for a 32K maximum machine. However, the disk drive's effect is to reduce overhead on the machine. Because it is so easy to load a file, you tend to keep everything on disk and only put the program or data file you need at the moment into your computer.

The floppy system is not perfect. One annoyance is that after each operation on disk you have to go back to the floppy menu. Thus, if you want to erase five files, you have to remove them one at a time, going back to the menu and choosing F5 (Erase) each time. It would be much nicer if you could call up the menu of files on the disk, highlight those you want to kill and then delete them all with one push of F5. It would also be nice if you could print files directly from disk instead of having to load them into the computer first. At least one programmer is working on a system to do this, but he reports that the floppy driver program is not designed to allow this to be done.

Another drawback of the system is that it does not allow random access inside files stored on disks. Thus, you cannot build and work in disks that you cannot load completely into your computer.

These problems, however, are relatively unimportant, and they can be fixed in future versions of the floppy program. The important thing is that the new drive is here, it is dependable and tough enough to stand up to travel, and it is worth every penny of its \$200 price. Still, I wouldn't throw away my tape recorder. Tapes are less expensive than disks, and are excellent for holding file backups.

(Tandy Corp., One Tandy Center, Fort Worth, TX 76102, \$200)

— G. Berton Latamore

The following products recently have been received by PCM, examined by our magazine staff and approved for the PCM Seal of Certification, your assurance that we have seen the product and have ascertained that is what it purports to be. This month the Seal of Certification has been issued to:

Due to a typographical error in the January issue of PCM (Page 93), the wrong price was given for *The Banner Machine* from Cardinal Software. *The Banner Machine* retails for \$149.95. We regret any inconvenience caused by this error.

10 Base, database management system designed for use with Fox Research's 10 Net local area network. Requires Tandy 1000, 1200 or 3000. Fox Research, Inc., 7005 Corporate Way, Dayton, OH 45459, (513) 433-2238, \$495.

10 Net, local area networking software and hardware. Based on Ethernet. One package required for each computer in the network. Requires Tandy 1000, 1200 or 3000. Fox Research, Inc., 7005 Corporate Way, Dayton, OH 45459, (513) 433-2238, \$695.

BDL.Celebrate, helps you keep track of special events such as birthdays, anniversaries and holidays. Program stores names and addresses and will print mailing labels. Requires Tandy 1000, 1200 or 3000. BDL Homeware, 2509 North Campbell Avenue, #328, Tucson, AZ 85719, (602) 577-1435, \$34.95.

BDL.Grow, medical records system for children. The software consists of four sections: Birth, Growth, Immunization and Doctor Name and Address. Requires Tandy 1000, 1200 or 3000. BDL Homeware, 2509 North Campbell Avenue, #328, Tucson, AZ 85719, (602) 577-1435, \$39.95.

BDL.Health, maintains family health history and medical records. Reports may be generated by family member in chronological order. Requires Tandy 1000, 1200 or 3000. BDL Homeware, 2509 North Campbell Avenue, #328, Tucson, AZ 85719, (602) 577-1435, \$39.95.

BDL.Roast, aids in the preparation of meat. Calculates cooking times and temperatures based on meat type and weight. Requires Tandy 1000, 1200 or 3000. BDL Homeware, 2509 North Campbell

Avenue, #328, Tucson, AZ 85719, (602) 577-1435, \$24.95.

EZTax-PLAN PRO, tax planning spreadsheet templates for use with Lotus 1-2-3. Templates also included for the new "Treasury II" tax simplification proposal. Requires Tandy 1000, 1200, 2000 or 3000 with Lotus 1-2-3. EZWare Corporation, P.O. Box 620, 29 Bala Avenue, Suite 206, Bala Cynwyd, PA 19004, (215) 667-4064, \$295, plus \$3 S/H.

Harvey ProDisk Control, allows you to create custom pop-up menus for controlling DOS and applications programs. Requires Tandy 1000, 1200 or 3000. Harvey, BHC Corporation, P.O. Box 1863, Cape Coral, FL 33910, (813) 482-8600, \$90.

PlayWriter Mystery!, software, paper and binding for making your own personalized book. A creative writing tool that reinforces reading, spelling, grammar and vocabulary skills. Requires Tandy 1000, 1200 or 3000. Woodbury, 127 White Oak Lane, CN1001, Old Bridge, NJ 08857, (201) 679-0200, \$39.95.

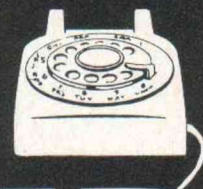
PROBALOTO, a program for picking lottery numbers. Requires Tandy 1000, 1200 or 3000. Gary Olander, 322 Haymarket Place, Gahanna, OH 43230, (614) 475-3315, \$29.95.

SuperBatch, an enhanced batch command processing utility that automatically executes DOS commands and passes parameters to applications programs. Requires Tandy 1000, 1200 or 3000. Merrill Street Software, 251 Merrill Street, Birmingham, MI 48011, (313) 645-5280, \$79.95.

Turbo Editor Toolbox, programming tools for developing customized full-screen editors for stand-alone use or for use in Turbo PASCAL programs. Requires Tandy 1000, 1200 or 3000 and Turbo PASCAL. Borland International, 4585 Scotts Valley Drive, Scotts Valley, CA 95066, (408) 438-8400, \$69.95.

By awarding a Seal, the magazine certifies the program does exist, but this does not constitute any guarantee of satisfaction. As soon as possible, these hardware or software items will be forwarded to PCM's reviewers for evaluation.

The Accessible Computer Experts



DELKER

HARDWARE SPECIALS

Tandy 600	CALL
Tandy 3000	CALL
25-1000 Tandy 1000 2-drive	\$ CALL
Tandy 1000 10MB Hd	CALL
26-3802 Model 100 24K	\$ 449.95
26-3860 Model 200	\$ 749.95
26-3816P 8K Upgrade Model 100	\$ 39.95
26-3127 64K Extended Coco 2	\$ 149.95
26-3131 Coco Drive	\$ 189.95
26-3124 Multi-Pac Interface	\$ 89.95
Video Monitor Adapters	\$ 34.95
Botek Ser/Par Interface	\$ 69.95
Extended Basic w/bk	\$ 39.95
64K (DEI) Memory upg	\$ 39.95
HJL Keyboard (D.E.F.2)	\$ 79.95
26-1276 DMP-105 Printer 80 cps	CALL
26-1278 DWP 220 Printer	CALL

ACCESSORIES

Volkmodem 1200	\$229.95
Novation J-Cat Modem	\$129.95
USR Password 300	\$179.95
Hayes SM 300 Modem	\$239.95
USR Password 1200	\$399.95
USR Password 2400	\$599.95
CoCo Switcher	\$ 39.95
Elephant Disks ssdd	\$ 18.95
26-3030 OS-9(64K)	\$ 64.95(disk)
Basic-09(req. OS-9)	\$ 87.95(disk)
"C" Compiler (OS-9)	\$ 87.95(disk)
FHL O-Pak (req. OS-9)	\$ 34.95(disk)
Elite Word	\$ 59.95(d&c)
Elite Calc	\$ 59.95(d&c)
Color Term Plus	\$ 29.95(cass)
26-3012 Deluxe Joystick	\$ 24.95
Tandy 1000 options	CALL

NEW! Dual Double Sides Drives including case, power supply & cable \$309.95

NEW! 26-3128 64K Direct Video CoCo2 \$219.95

MSI SOFTWARE

MSI DISKUTIL	NEW	\$ 19.95
COLOR FINANCE II	NEW	\$ 69.95
MSI NAMEFILE		\$ 24.95
MSI CALENDAR	NEW	\$ 19.95
MSI COLOR PAYROLL	NEW	\$ 99.95
COLOR FINANCE 4		\$ 69.95

TRS-80 Trademark Tandy Corporation. Prices subject to change without notice. Write for our FREE newsletter.

Call for prices and availability of your favorite software and hardware. All advertised items subject to availability. Prices do not include shipping and handling. All of the above units are covered by our 120 day carry-in warranty.



DELKER ELECTRONICS, INC.
P.O. Box 897 408 C Nissan Blvd.
Smyrna, TN 37167

■ 800-251-5008 ■ 615-459-2636 (Tennessee)
■ 800-545-2502 ■ 615-254-0088 (Tennessee) (Nashville)

Visa, Mastercard and American Express

Using *BAREAD* 2.1

Bar code listings must be read in numerical order beginning with Line 1 and continuing through the last line of the listing. The computer display is used to prompt you as to which line to scan and give you warning messages should you happen to get out of step.

When you run *BAREAD*, it asks you to scan the first line of the bar code listing. This line contains the name of the program as well as the beginning of the program itself. The computer will sound a high-pitched beep whenever it's ready for you to scan a line. After a line has been successfully read, you'll hear a lower beep. A "blip-bloop" sound prompts you to turn your attention to the screen for a message. You'll hear this when you accidentally scan a line out of sequence.

After reading the first line, you continue scanning with

the second line. Remember to wait for a high beep before scanning and then listen for a low beep to indicate a successful read.

Once the last line of the listing has been scanned, *BAREAD* will return control to the Tandy 100/200 menu screen. Note that the program you just scanned is now in the directory with a .DD extension.

The final step is to convert the .DD text file to a normal BASIC program. This is done quite simply by going to BASIC and loading the file with a command such as `LOAD"TEST.DD"` (if the program name were TEST). The program will load into BASIC and will be ready to run. To save the program in BASIC's compressed format (.BA extension), you'd type `SAVE"TEST"` (if the program were named TEST). You may then kill the .DD file with `KILL "TEST.DD"`.

BAREAD 2.1

```

1000 ' *** Initialize ***
1010 ON ERROR GOTO 1040
1020 CLEAR 1000:MAXFILES=2
1030 GOTO 1050
1040 IF ERR=5 THEN RESUME NEXT
1050 ON ERROR GOTO 0
1060 RUNM "B3OF9"
1070 OPEN "WAND:" FOR INPUT AS #1
1080 UC$=-1
1090 PC$="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ- $+"
1100 DIM RW$(36)
1110 ER$(1)="You must scan line 1 first!"
1120 ER$(2)="You've SKIPPED a line!"
1130 ER$(3)="You've ALREADY SCANNED this line!"
1140 ER$(4)="Code not PCM2/39 format!"
1150 ER$(5)="Command not applicable here!"
1160 ER$(6)="You cannot skip this line!"
1170 ER$(7)="Selected resume file not in computer!"
1180 ' *** Read Reserved Words List ***
1190 DATA BEEP,CLEAR,CLOSE,DATA,DEFDBL,DEFINT,DEFPROC,DEFSTR,ELSE,GOSUB,GOTO
1200 DATA INKEY$,INPUT,INSTR(,LCOPY,LEFT$(,LINE(,LOADM,LPRINT,USING,MAXFILES
1210 DATA MID$(,NEXT,PEEK,POKE,POWER,PRESERVE,PRINT,READ,RESTORE,RETURN,RIGHT$(
1220 DATA SOUND,SPACE$(,STRING$(,THEN
1230 FOR I%=1 TO 36:READ RW$(I%):NEXT I%
1240 ' *** Procedure Begins Here ***
1250 CLS:PRINT@44,"PCM Bar Code Program Reader v2.1"
1260 LINE(20,4)-(219,18),1,B:LINE(22,6)-(217,16),1,B
1270 NN%=1
1280 GOSUB 1660:IF ER%>0 THEN GOSUB 1620:GOTO 1280
1290 IF LL%=0 AND INSTR("YN",IL$)>0 THEN ER%=5:GOSUB 1620:GOTO 1280
1300 IF LL%=0 THEN ON INSTR("ALSR",IL$) GOTO 1820,1890,1980,2050
1310 IF LL%=1295 THEN 1350
1320 IF LL%<>NN% AND NN%=1 THEN ER%=1:GOSUB 1620:GOTO 1280

```

```

1330 IF LL%<NN% THEN ER%=3:GOSUB 1620:GOTO 1280
1340 IF LL%>NN% AND NN%>1 THEN ER%=2:GOSUB 1620:GOTO 1280
1350 IL$=RIGHT$(IL$,19)
1360 IF LL%=1 AND NN%>0 THEN GOSUB 1780
1370 CL$=CL$+IL$
1380 FOR I%=1 TO LEN(CL$)
1390   CH$=MID$(CL$,I%,1)
1400   IF CH$="%" THEN GOSUB 1510:IF NL% THEN 1470 ELSE GOTO 1440
1410   IF CH$="/" THEN GOSUB 1550:IF NL% THEN 1470 ELSE GOTO 1440
1420   IF CH$="." THEN UC%=NOT(UC%):GOTO 1450
1430   IF CH$="A" AND CH$<="Z" AND NOT(UC%) THEN CH$=CHR$(ASC(CH$)+32)
1440   XX$=XX$+CH$:IF RIGHT$(XX$,1)=CHR$(13) THEN PRINT#2,XX$;:XX$="":UC%=-1
1450 NEXT I%
1460 CL$=""
1470 PRINT@200,SPACE$(80);
1480 IF LL%<>1295 THEN NN%=LL%+1:GOTO 1280
1490 ' *** Done ***
1500 CLOSE:CALL 61807!:CLEAR 500,HIMEM:MENU
1510 ' *** Decode Reserved Word ***
1520 NL%=0:IF I%>LEN(CL$)-1 THEN NL%=-1:CL$="":GOTO 1540
1530 I%=I%+1:CH$=RW$(INSTR(PC$,MID$(CL$,I%,1)))
1540 RETURN
1550 ' *** Decode Hex and Control Characters ***
1560 NL%=0:IF I%>LEN(CL$)-1 THEN NL%=-1:CL$="":GOTO 1610
1570 I%=I%+1:IF INSTR("/%.",MID$(CL$,I%,1))>0 THEN CH$=MID$(CL$,I%,1):GOTO 1610
1580 IF I%>LEN(CL$)-1 THEN NL%=-1:CL$=RIGHT$(CL$,2):GOTO 1610
1590 HX$=MID$(CL$,I%,2):CH$=CHR$((INSTR("0123456789ABCDEF",LEFT$(HX$,1))-1)*16+INSTR("0123456789ABCDEF",RIGHT$(HX$,1))-1)
1600 I%=I%+1
1610 RETURN
1620 ' *** Error Codes ***
1630 SOUND 5000,10:SOUND 8000,10:SOUND 5000,10
1640 PRINT@220-.5*LEN(ER$(ER%)),ER$(ER%);
1650 RETURN
1660 ' *** Get Code Line ***
1670 PRINT@173,"";:PRINT USING "Scan line ###";NN%
1680 IF NN%=-1 THEN PRINT@173,"Scan any line":GOTO 1700

```

































```

1690 SOUND 500,5
1700 INPUT#1,IL$:ER%=0
1710 FOR I%=1 TO LEN(IL$)
1720 IF MID$(IL$,I%,1)="!" THEN MID$(IL$,I%,1)=". "
1730 NEXT I%
1740 IF LEN(IL$)<>1 AND LEN(IL$)<>21 THEN ER%=4:RETURN
1750 IF LEN(IL$)=1 THEN LL%=0:RETURN
1760 LL$=LEFT$(IL$,2):LL%=(INSTR("0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ",LEFT$(LL$,1))-1)*36+INSTR("0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ",RIGHT$(LL$,1))-1
1770 RETURN
1780 ' *** Open Program File ***
1790 PN$=LEFT$(IL$,6):IL$=RIGHT$(IL$,LEN(IL$)-6)
1800 OPEN PN$ FOR OUTPUT AS #2
1810 RETURN
1820 ' *** Abort ***
1830 BEEP:BEEP:BEEP
1840 PRINT@209,"ABORT! Are you sure?";
1850 INPUT#1,AN$
1860 IF INSTR("YN",AN$)=0 THEN BEEP:PRINT@251,"Scan 'YES' or 'NO'":GOTO 1850
1870 PRINT@200,SPACE$(80);
1880 IF AN$="Y" THEN CLOSE:KILL PN$+".DO":GOTO 1490 ELSE GOTO 1280
1890 ' *** Skip Line ***
1900 IF NN%=1 THEN ER%=6:GOSUB 1620:GOTO 1280
1910 BEEP:BEEP:BEEP

1920 PRINT@210,"SKIP! Are you sure?"
1930 INPUT#1,AN$
1940 IF INSTR("YN",AN$)=0 THEN BEEP:PRINT@251,"Scan 'YES' or 'NO'":GOTO 1930
1950 PRINT@200,SPACE$(80);
1960 IF AN$="Y" THEN NN%=NN%+1
1970 GOTO 1280
1980 ' *** Stop & Save ***
1990 BEEP:BEEP:BEEP
2000 PRINT@207,"STOP & SAVE! Are you sure?";
2010 INPUT#1,AN$
2020 IF INSTR("YN",AN$)=0 THEN BEEP:PRINT@251,"Scan 'YES' or 'NO'":GOTO 2010
2030 PRINT@200,SPACE$(80);
2040 IF AN$="Y" THEN 1490 ELSE GOTO 1280
2050 ' *** Resume ***
2060 IF NN%<>1 THEN ER%=5:GOSUB 1620:GOTO 1280
2070 PRINT@254,"Resume Mode";
2080 NN%=1:GOSUB 1660
2090 IF LL%=0 THEN ER%=5 ELSE IF LL%<>1 THEN ER%=1
2100 IF ER%>0 THEN GOSUB 1620:GOTO 2060
2110 PN$=MID$(IL$,3,6)
2120 ON ERROR GOTO 2140
2130 OPEN PN$ FOR INPUT AS #2:GOTO 2170
2140 RESUME 2150
2150 CLOSE #2
2160 ER%=7:GOSUB 1620:GOTO 1270
2170 CLOSE #2:OPEN PN$ FOR APPEND AS #2
2180 NN%=-1:GOTO 1280

```

CHILL (FROM PAGE 36)

1		14	
2		15	
3		16	
4		17	
5		18	
6		19	
7		20	
8		21	
9		22	
10		23	
11		24	
12		25	
13		26	
		27	
		28	
		29	
		30	

LOADER (FROM PAGE 78)

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

HEX.DO (FROM PAGE 78)

1

2

3

4

5

6

7

8

9

10

11

12

13

14

Abort

Skip Line

Stop & Save

Resume

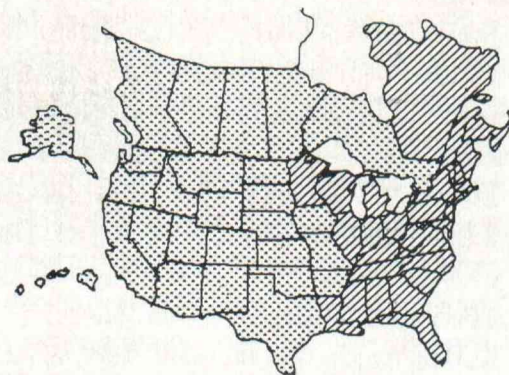
Yes

No

Also available at selected B.Dalton Bookseller stores

TNT SOFTWARE	72
TAB CO.	110
TLM SYSTEMS	65
TRUE DATA	77
WIZARD	110

PCM Advertising and Marketing Representatives



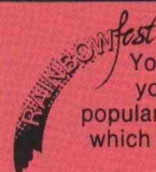
Kim Vincent
Advertising Representative
The Falsoft Building
P.O. Box 385
Prospect, KY 40059
(502) 228-4492

PCM

Palo Alto

fest

Feb. 14-16



Your admission to PCMfest also entitles you to attend RAINBOWfest, the highly popular show for the Tandy Color Computer which will run concurrently with PCMfest at the same location.

You're invited to PCMfest, a great show that's just for your Tandy computer. Sponsored by PCM, *The Personal Computer Magazine for Tandy Computer Users*, it's a wonderful way to meet and exchange information with those who share your interest in the new generation of Tandy computers.

You'll also discover the greatest variety of products ever for your computer because all of the favorites and a lot of new ones will be featured in the exhibits of PCM advertisers. Try out that new program and take it home that very day!

Along with other PCM readers, you'll meet the top national experts on your computer, including those who write for or who are written about in PCM. They will answer your questions on the spot.

PCMfest also will include a comprehensive lineup of free seminars on topics of immediate concern — and all of them designed to help you get the most out of your Tandy computer.

The Hyatt Hotels-Palo Alto will be offering special rates (\$71, single or double room) for PCMfest. The show opens Friday evening with a 7 p.m. to 10 p.m. session. It's a daytime-only show Saturday — the exhibits open at 10 a.m. and run continuously until 6 p.m. On Sunday, the exhibit hall opens at 11 a.m. and closes at 4 p.m.

Tickets may be obtained directly from PCM. We'll also send you a special reservation form so you can take advantage of the special room rate. Come to PCMfest and

let's celebrate the new generation of Tandy computers!

Keynote Speaker

William D. Gattis

Vice President of the Education Division of Radio Shack will be the keynote speaker of the Community Breakfast, Saturday.

Free Seminars

Sam Redmon

The Future of Portable Computing

As co-founder of the Dallas-based Portable Computer Support Group, Sam markets a wide variety of software and hardware for the Tandy portables.

Bill Barden

Assembly Language

Respected author of 30 books on various computer subjects, Bill will speak on one of his specialties, assembly language programming.

Steve Bjork

User Interfaces

Owner of SRB Software and author of numerous commercial software products, Steve will speak on creating better user interfaces.

Howard Wolowitz

Database Management

President of Small Computer Company, Howard is one of the developers of Small's popular *Profile* and *filePro* database management systems.

Danny Humphress

ViaNet — Tandy's Intra-Office Communications System

PCM managing editor and owner of a consulting firm in Louisville, Kentucky, Danny will take away the mysteries of using Tandy's new ViaNet networking system.

YES, I'm coming to Palo Alto! I want to save by buying tickets now at the special advance sale price.

Please send me:

_____ three-day tickets at \$9 each total _____
_____ one-day tickets at \$7 each total _____
(Circle one) Friday Saturday Sunday
_____ Saturday Breakfast at \$12 each total _____
Handling Charge \$1 _____

TOTAL ENCLOSED _____

(U.S. Currency Only, Please)

☐ Also send me a hotel reservation card for the Hyatt Hotels-Palo Alto (\$71, single or double room).

Make checks payable to: PCM. Mail to: PCMfest, The Falsoft Building, P.O. Box 385, Prospect, KY 40059. To make reservations by phone, call: (502) 228-4492.

Advance ticket deadline: Feb. 7, 1986. Orders received less than two weeks prior to show opening will be held for you at the door. Tickets will also be available at the door at a slightly higher price. Children under four, free; four and over, full admission price.

Name _____
Address _____
City _____ State _____
Telephone _____ ZIP _____
Company _____
☐ Payment enclosed, OR ☐ Charge to:
☐ VISA ☐ MasterCard ☐ American Express
Account Number _____
Exp. Date _____
Signature _____

SUPER ROM

Lucid Spreadsheet Write ROM Database Outliner



NOW YOU CAN REALLY HAVE IT ALL!

All on one ROM. Truly the finest four programs available for the Model 100 — guaranteed. Try it for 30 days. If you aren't blown away by the excellence return it for a full refund.

\$199⁹⁵

The four best programs for the Model 100 all on one ROM. 32K of power without using any RAM for program storage. This is the PCSG Snap-In ROM that just presses easily into the little ROM socket in the compartment on the back. You access the four right from the main menu like built-ins.

Write ROM — the definitive word processor for the Model 100. Function key formatting or dot commands. Search and replace. Library feature — inserts words, phrases or whole documents into text from just a code. MAP lets you see a picture of your document. In all there are 60 features and functions. No one can claim faster operation. FORM lets you create interactive forms with on-screen prompts that you can answer from the keyboard. Nothing else for the Model 100 compares with the features of Write ROM. Exactly the same as the Write ROM sold as a single program. Infoworld says it "makes the Model 100 a viable writing unit ... sur-

passed our highest expectations for quality and clarity."

Lucid Spreadsheet: This is the one PICO magazine says "blows Multiplan right out of the socket" and Infoworld performance rated as "excellent" and said "makes the Model 100 compute." Gives you features you cannot get with Lotus 123. Lets you build spreadsheets in your Model 100 that would consume 140-150K on a desktop. Program generating capability with no programming knowledge required. Variable column widths. Includes find and sort with function key control. It's fast, recalculates like lightning. No feature has been taken from the original, only new ones added.

Database: This is a relational data base like no other. You can do everything from mailing lists to invoices. No complicated pseudo-coding, you create input screens as simply as typing into TEXT. You are not limited by size; you can have as large an input screen as you wish. Prints out reports or forms, getting information from as many files as

you like. Complete math between fields. Total interface with Lucid worksheets.

Outliner: Does everything that Think-tank does on a PC but a whole lot better. Includes a Sort for your headlines. Lets you have headlines of up to 240 characters. Has cloning, hoisting and sideways scroll up to 250 characters. Like Lucid, this one sets a new standard for outliners. This is the way to plan and organize your projects.

Present Lucid and Write ROM owners can upgrade for \$100. If you have both it's \$75.

As usual PCSG sells the Super ROM on a thirty day guarantee. If for any reason you are not satisfied, simply return it for a full refund.

We are excited about this product. Super ROM gives the Model 100 the true power of a desktop. No other multi-program ROM has software that compares. But don't take our word for it. We invite you to make that comparison yourself. Priced at \$199.95 on Snap-In ROM.

Got stuck with somebody else's multi-ROM? We'll upgrade it for \$100.

(214) 351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd., #206, Dallas, TX 75229

© PCSG

MC, Visa, American Express, Check, or C.O.D.